10/031919

Copy for the Elected Office (EO/US)

PCT/DK00/00354

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION OF THE RECORDING OF A CHANGE

(PCT Rule 92bis.1 and Administrative Instructions, Section 422)

To:

PLOUGMANN, VINGTOFT & PARTNERS A/S
Sankt Annæ Plads 11
DK-1250 Copenhagen K
DANEMARK

Date of mailing (day/month/year)
31 January 2002 (31.01.02)

Applicant's or agent's file reference
25107 PC 1

IMPORTANT NOTIFICATION

International application No.
PCT/DK00/00354

International filing date (day/month/year)
30 June 2000 (30.06.00)

1. The following indications appeared on record concerning:

[X] the applicant [ ] the inventor [ ] the agent [ ] the common representative

Name and Address
MINDPASS A/S
Vardevej 1
DK-9220 Aalborg ø
Denmark

State of Nationality: DK
State of Residence: DK
Telephone No.:
Facsimile No.:
Teleprinter No.:

2. The International Bureau hereby notifies the applicant that the following change has been recorded concerning:

[ ] the person [ ] the name [ ] the address [ ] the nationality [ ] the residence

Name and Address

State of Nationality:
State of Residence:
Telephone No.:
Facsimile No.:
Teleprinter No.:

3. Further observations, if necessary:
The above-mentioned applicant has been removed from our records. PEDERSEN, Torbe, Bach, JENSEN, Christian, S. and DYRESON, Curtis, E. are now the sole joint applicants/inventors for all designated States.

4. A copy of this notification has been sent to:

[X] the receiving Office
[ ] the International Searching Authority
[ ] the International Preliminary Examining Authority
[ ] the designated Offices concerned
[X] the elected Offices concerned
[X] other: MINDPASS A/S

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No.: (41-22) 740.14.35

Authorized officer
François BAECHLER

Telephone No.: (41-22) 338.83.38

Form PCT/IB/306 (March 1994)

004634275

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION OF ELECTION

(PCT Rule 61.2)

To:

Commissioner
US Department of Commerce
United States Patent and Trademark Office, PCT
2011 South Clark Place Room
CP2/5C24
Arlington, VA 22202
ETATS-UNIS D'AMERIQUE

in its capacity as elected Office

Date of mailing (day/month/year) 23 March 2001 (23.03.01)	
International application No. PCT/DK00/00354	**Applicant's or agent's file reference** 25107 PC 1
International filing date (day/month/year) 30 June 2000 (30.06.00)	**Priority date** (day/month/year) 21 July 1999 (21.07.99)
Applicant PEDERSEN, Torben, Bach et al	

1. The designated Office is hereby notified of its election made:

   [X] in the demand filed with the International Preliminary Examining Authority on:

   29 January 2001 (29.01.01)

   [ ] in a notice effecting later election filed with the International Bureau on:

2. The election [X] was

   [ ] was not

   made before the expiration of 19 months from the priority date or, where Rule 32 applies, within the time limit under Rule 32.2(b).

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland

Facsimile No.: (41-22) 740.14.35

Authorized officer

Charlotte ENGER

Telephone No.: (41-22) 338.83.38

10/031911 VOR

PCT

REQUEST

The undersigned requests that the present international application be processed according to the Patent Cooperation Treaty.

For receiving Office use only

International Application No.

International Filing Date

Name of receiving Office and "PCT International Application"

Applicant s or agent s file reference *(if desired) (12 characters maximum)* 25107 PC 1

Box No. I TITLE OF INVENTION

Method and systems for making OLAP hierarchies summerisable

Box No. II APPLICANT

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

Mindpass A/S
Vardevej 1
DK-9220 Aalborg Ø
Denmark

☐ This person is also inventor.

Telephone No.

Facsimile No.

Teleprinter No.

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

This person is applicant for the purposes of: ☐ all designated States ☒ all designated States except the United States of America ☐ the United States of America only ☐ the States indicated in the Supplemental Box

Box No. III FURTHER APPLICANT(S) AND/OR (FURTHER) INVENTOR(S)

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

PEDERSEN, Torben Bach
Trøjborgvej 32, 2. th.
DK-8200 Århus N
Denmark

This person is:

☐ applicant only

☒ applicant and inventor

☐ inventor only *(If this check-box is marked, do not fill in below.)*

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

This person is applicant for the purposes of: ☒ all designated States ☐ all designated States except the United States of America ☐ the United States of America only ☐ the States indicated in the Supplemental Box

☒ Further applicants and/or (further) inventors are indicated on a continuation sheet.

Box No. IV AGENT OR COMMON REPRESENTATIVE; OR ADDRESS FOR CORRESPONDENCE

The person identified below is hereby/has been appointed to act on behalf of the applicant(s) before the competent International Authorities as: ☒ agent ☐ common representative

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

Plougmann, Vingtoft & Partners A/S
Sankt Annæ Plads 11
DK-1250 Copenhagen K
Denmark

Telephone No.
+45 33 63 93 00

Facsimile No.
+45 33 63 96 00

Teleprinter No.
-

☐ **Address for correspondence:** Mark this check-box where no agent or common representative is/has been appointed and the space above is used instead to indicate a special address to which correspondence should be sent.

Continuation of Box No. III **FURTHER APPLICANT(S) AND/OR (FURTHER) INVENTOR(S)**

If none of the following sub-boxes is used, this sheet should not be included in the request.

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

JENSEN, Christian S.
Fr. Bajers Vej 70
DK-9220 Aalborg Ø
Denmark

This person is:
- ☐ applicant only
- ☒ applicant and inventor
- ☐ inventor only *(If this check-box is marked, do not fill in below.)*

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

This person is applicant for the purposes of:
- ☒ all designated States
- ☐ all designated States except the United States of America
- ☐ the United States of America only
- ☐ the States indicated in the Supplemental Box

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

DYRESON, Curtis E.
6/180 Ron Penhaligon Drive
Robina, Queensland 4226
Australia

This person is:
- ☐ applicant only
- ☒ applicant and inventor
- ☐ inventor only *(If this check-box is marked, do not fill in below.)*

State *(that is, country)* of nationality: US

State *(that is, country)* of residence: AU

This person is applicant for the purposes of:
- ☒ all designated States
- ☐ all designated States except the United States of America
- ☐ the United States of America only
- ☐ the States indicated in the Supplemental Box

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

This person is:
- ☐ applicant only
- ☐ applicant and inventor
- ☐ inventor only *(If this check-box is marked, do not fill in below.)*

State *(that is, country)* of nationality:

State *(that is, country)* of residence:

This person is applicant for the purposes of:
- ☐ all designated States
- ☐ all designated States except the United States of America
- ☐ the United States of America only
- ☐ the States indicated in the Supplemental Box

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country. The country of the address indicated in this Box is the applicant s State (that is, country) of residence if no State of residence is indicated below.)*

This person is:
- ☐ applicant only
- ☐ applicant and inventor
- ☐ inventor only *(If this check-box is marked, do not fill in below.)*

State *(that is, country)* of nationality:

State *(that is, country)* of residence:

This person is applicant for the purposes of:
- ☐ all designated States
- ☐ all designated States except the United States of America
- ☐ the United States of America only
- ☐ the States indicated in the Supplemental Box

☐ Further applicants and/or (further) inventors are indicated on another continuation sheet.

Box No.V DESIGNATION OF STATES

The following designations are hereby made under Rule 4.9(a) *(mark the applicable check-boxes; at least one must be marked)*:

Regional Patent

☒ **AP** **ARIPO Patent:** **GH** Ghana, **GM** Gambia, **KE** Kenya, **LS** Lesotho, **MW** Malawi, **SD** Sudan, **SL** Sierra Leone, **SZ** Swaziland, **TZ** United Republic of Tanzania, **UG** Uganda, **ZW** Zimbabwe, and any other State which is a Contracting State of the Harare Protocol and of the PCT

☒ **EA** **Eurasian Patent:** **AM** Armenia, **AZ** Azerbaijan, **BY** Belarus, **KG** Kyrgyzstan, **KZ** Kazakhstan, **MD** Republic of Moldova, **RU** Russian Federation, **TJ** Tajikistan, **TM** Turkmenistan, and any other State which is a Contracting State of the Eurasian Patent Convention and of the PCT

☒ **EP** **European Patent:** **AT** Austria, **BE** Belgium, **CH and LI** Switzerland and Liechtenstein, **CY** Cyprus, **DE** Germany, **DK** Denmark, **ES** Spain, **FI** Finland, **FR** France, **GB** United Kingdom, **GR** Greece, **IE** Ireland, **IT** Italy, **LU** Luxembourg, **MC** Monaco, **NL** Netherlands, **PT** Portugal, **SE** Sweden, and any other State which is a Contracting State of the European Patent Convention and of the PCT

☒ **OA** **OAPI Patent:** **BF** Burkina Faso, **BJ** Benin, **CF** Central African Republic, **CG** Congo, **CI** Côte d Ivoire, **CM** Cameroon, **GA** Gabon, **GN** Guinea, **GW** Guinea-Bissau, **ML** Mali, **MR** Mauritania, **NE** Niger, **SN** Senegal, **TD** Chad, **TG** Togo, and any other State which is a member State of OAPI and a Contracting State of the PCT *(if other kind of protection or treatment desired, specify on dotted line)* .....

National Patent *(if other kind of protection or treatment desired, specify on dotted line)*:

☒ **AE** United Arab Emirates
☒ **AL** Albania
☒ **AM** Armenia
☒ **AT** Austria and utility model
☒ **AU** Australia
☒ **AZ** Azerbaijan
☒ **BA** Bosnia and Herzegovina
☒ **BB** Barbados
☒ **BG** Bulgaria
☒ **BR** Brazil
☒ **BY** Belarus
☒ **CA** Canada
☒ **CH and LI** Switzerland and Liechtenstein
☒ **CN** China
☒ **CR** Costa Rica
☒ **CU** Cuba
☒ **CZ** Czech Republic and utility model
☒ **DE** Germany and utility model
☒ **DK** Denmark and utility model
☒ **DM** Dominica
☒ **EE** Estonia and utility model
☒ **ES** Spain
☒ **FI** Finland and utility model
☒ **GB** United Kingdom
☒ **GD** Grenada
☒ **GE** Georgia
☒ **GH** Ghana
☒ **GM** Gambia
☒ **HR** Croatia
☒ **HU** Hungary
☒ **ID** Indonesia
☒ **IL** Israel
☒ **IN** India
☒ **IS** Iceland
☒ **JP** Japan
☒ **KE** Kenya
☒ **KG** Kyrgyzstan
☒ **KP** Democratic People s Republic of Korea
☒ **KR** Republic of Korea and utility model
☒ **KZ** Kazakhstan
☒ **LC** Saint Lucia
☒ **LK** Sri Lanka
☒ **LR** Liberia
☒ **LS** Lesotho
☒ **LT** Lithuania
☒ **LU** Luxembourg
☒ **LV** Latvia
☒ **MA** Morocco
☒ **MD** Republic of Moldova
☒ **MG** Madagascar
☒ **MK** The former Yugoslav Republic of Macedonia
☒ **MN** Mongolia
☒ **MW** Malawi
☒ **MX** Mexico
☒ **NO** Norway
☒ **NZ** New Zealand
☒ **PL** Poland
☒ **PT** Portugal
☒ **RO** Romania
☒ **RU** Russian Federation
☒ **SD** Sudan
☒ **SE** Sweden
☒ **SG** Singapore
☒ **SI** Slovenia
☒ **SK** Slovakia and utility model
☒ **SL** Sierra Leone
☒ **TJ** Tajikistan
☒ **TM** Turkmenistan
☒ **TR** Turkey
☒ **TT** Trinidad and Tobago
☒ **TZ** United Republic of Tanzania
☒ **UA** Ukraine
☒ **UG** Uganda
☒ **US** United States of America
☒ **UZ** Uzbekistan
☒ **VN** Viet Nam
☒ **YU** Yugoslavia
☒ **ZA** South Africa
☒ **ZW** Zimbabwe

Check-boxes reserved for designating States which have become party to the PCT after issuance of this sheet:

☒ DZ Algeria
☒ AG Antigua and Barbuda
☒ MZ Mozambique
☒ BZ Belize

Precautionary Designation Statement: In addition to the designations made above, the applicant also makes under Rule 4.9(b) all other designations which would be permitted under the PCT except any designation(s) indicated in the Supplemental Box as being excluded from the scope of this statement. The applicant declares that those additional designations are subject to confirmation and that any designation which is not confirmed before the expiration of 15 months from the priority date is to be regarded as withdrawn by the applicant at the expiration of that time limit. *(Confirmation (including fees) must reach the receiving Office within the 15-month time limit.)*

Box No. VI PRIORITY CLAIM ☐ Further priority claims are indicated in the Supplemental Box.

Filing date of earlier application (day/month/year)	Number of earlier application	Where earlier application is: national application: country	regional application:* regional Office	international application: receiving Office
item (1) 21 July 1999 (21/07/1999)	PA 1999 01045	DK		
item (2)				
item (3)				

☒ The receiving Office is requested to prepare and transmit to the International Bureau a certified copy of the earlier application(s) *(only if the earlier application was filed with the Office which for the purposes of the present international application is the receiving Office)* identified above as item(s): (1)

** Where the earlier application is an ARIPO application, it is mandatory to indicate in the Supplemental Box at least one country party to the Paris Convention for the Protection of Industrial Property for which that earlier application was filed (Rule 4.10(b)(ii)). See Supplemental Box.*

Box No. VII INTERNATIONAL SEARCHING AUTHORITY

Choice of International Searching Authority (ISA) *(if two or more International Searching Authorities are competent to carry out the international search, indicate the Authority chosen; the two-letter code may be used)* : ISA / EP

Request to use results of earlier search; reference to that search *(if an earlier search has been carried out by or requested from the International Searching Authority):*

Date *(day/month/year)* Number Country *(or regional Office)*

Box No. VIII CHECK LIST; LANGUAGE OF FILING

This international application contains the following **number of sheets**:

request	:	4
description (excluding sequence listing part)	:	48
claims	:	9
abstract	:	1
drawings	:	4
sequence listing part of description	:	-
Total number of sheets	:	66

This international application is **accompanied by** the item(s) marked below:

1. ☒ fee calculation sheet
2. ☐ separate signed power of attorney
3. ☐ copy of general power of attorney; reference number, if any:
4. ☐ statement explaining lack of signature
5. ☐ priority document(s) identified in Box No. VI as item(s):
6. ☐ translation of international application into (language):
7. ☐ separate indications concerning deposited microorganism or other biological material
8. ☐ nucleotide and/or amino acid sequence listing in computer readable form
9. ☒ other *(specify)*: assignments, proving the applicants' right to claim priority

Figure of the drawings which should accompany the abstract:

Language of filing of the international application: English

Box No. IX SIGNATURE OF APPLICANT OR AGENT

Next to each signature, indicate the name of the person signing and the capacity in which the person signs (if such capacity is not obvious from reading the request).

Plougmann, Vingtoft & Partners A/S
30 June 2000

JJS

Jens Jørgen Schmidt

For receiving Office use only

1. Date of actual receipt of the purported international application:
2. Drawings: ☐ received: ☐ not received:
3. Corrected date of actual receipt due to later but timely received papers or drawings completing the purported international application:
4. Date of timely receipt of the required corrections under PCT Article 11(2):
5. International Searching Authority (if two or more are competent): ISA /
6. ☐ Transmittal of search copy delayed until search fee is paid.

For International Bureau use only

Date of receipt of the record copy by the International Bureau:

PATENT COOPERATION TREATY

From the INTERNATIONAL BUREAU

PCT

NOTIFICATION CONCERNING SUBMISSION OR TRANSMITTAL OF PRIORITY DOCUMENT

(PCT Administrative Instructions, Section 411)

To:

PLOUGMANN, VINGTOFT & PARTNERS A/S
Sankt Annæ Plads 11
DK-1250 Copenhagen K
DANEMARK

PLOUGMANN VINGTOFT & PARTNERS
08 DEC. 2000
JTS/MHO

Date of mailing (day/month/year) 17 November 2000 (17.11.00)	
Applicant's or agent's file reference 25107 PC 1	**IMPORTANT NOTIFICATION**
International application No. PCT/DK00/00354	International filing date (day/month/year) 30 June 2000 (30.06.00)
International publication date (day/month/year) Not yet published	Priority date (day/month/year) 21 July 1999 (21.07.99)
Applicant MINDPASS A/S et al	

1. The applicant is hereby notified of the date of receipt (except where the letters "NR" appear in the right-hand column) by the International Bureau of the priority document(s) relating to the earlier application(s) indicated below. Unless otherwise indicated by an asterisk appearing next to a date of receipt, or by the letters "NR", in the right-hand column, the priority document concerned was submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b).

2. This updates and replaces any previously issued notification concerning submission or transmittal of priority documents.

3. An **asterisk(*)** appearing next to a date of receipt, in the right-hand column, denotes a priority document submitted or transmitted to the International Bureau but not in compliance with Rule 17.1(a) or (b). In such a case, **the attention of the applicant is directed** to Rule 17.1(c) which provides that no designated Office may disregard the priority claim concerned before giving the applicant an opportunity, upon entry into the national phase, to furnish the priority document within a time limit which is reasonable under the circumstances.

4. The **letters "NR"** appearing in the right-hand column denote a priority document which was not received by the International Bureau or which the applicant did not request the receiving Office to prepare and transmit to the International Bureau, as provided by Rule 17.1(a) or (b), respectively. In such a case, **the attention of the applicant is directed** to Rule 17.1(c) which provides that no designated Office may disregard the priority claim concerned before giving the applicant an opportunity, upon entry into the national phase, to furnish the priority document within a time limit which is reasonable under the circumstances.

Priority date	Priority application No.	Country or regional Office or PCT receiving Office	Date of receipt of priority document
21 July 1999 (21.07.99)	PA 1999 01045	DK	15 Augu 2000 (15.08.00)

The International Bureau of WIPO
34, chemin des Colombettes
1211 Geneva 20, Switzerland
Facsimile No. (41-22) 740 14 35

Authorized officer

S. De Michiel

Telephone No. (41-22) 338.83.38

PLOUGMANN, VINGTOFT & PARTNERS

VOR

KOPI

The demand must be filed directly with the competent International Preliminary Examining Authority or, if two or more Authorities are competent, with the one chosen by the applicant. The full name or two-letter code of that Authority may be indicated by the applicant on the line below:

IPEA/ EP

PCT

CHAPTER II

DEMAND

under Article 31 of the Patent Cooperation Treaty:
The undersigned requests that the international application specified below be the subject of international preliminary examination according to the Patent Cooperation Treaty and hereby elects all eligible States (except where otherwise indicated).

For International Preliminary Examining Authority use only

Identification of IPEA	Date of receipt of DEMAND

Box No. I IDENTIFICATION OF THE INTERNATIONAL APPLICATION

Applicant's or agent's file reference: 25107 PC 1

International application No.	International filing date *(day/month/year)*	(Earliest) Priority date *(day/month/year)*
PCT/DK00/00354	30 June 2000 (30/06/2000)	21 July 1999 (21/07/1999)

Title of invention:
Method and systems for making OLAP hierarchies summerisable

Box No. II APPLICANT(S)

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

Mindpass A/S
Vardevej 1
DK-9220 Aalborg Ø
Denmark

Telephone No.:

Facsimile No.:

Teleprinter No.:

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

PEDERSEN, Torben Bach
Trøjborgvej 32, 2. th.
DK-8200 Århus N
Denmark

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

JENSEN, Christian S.
Fr. Bajers Vej 70
DK-9220 Aalborg Ø
Denmark

State *(that is, country)* of nationality: DK

State *(that is, country)* of residence: DK

[X] Further applicants are indicated on a continuation sheet.

International application No.
PCT/DK00/00354

Continuation of Box No. II APPLICANT(S)

If none of the following sub-boxes is used, this sheet should not be included in the demand.

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

DYRESON, Curtis E.
6/180 Ron Penhaligon Drive
Robina, Queensland 4226
Australia

State *(that is, country)* of nationality:	State *(that is, country)* of residence:
US	AU

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

State *(that is, country)* of nationality:	State *(that is, country)* of residence:

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

State *(that is, country)* of nationality:	State *(that is, country)* of residence:

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

State *(that is, country)* of nationality:	State *(that is, country)* of residence:

☐ Further applicants are indicated on another continuation sheet.

International application No.
PCT/DK00/00354

Box No. III AGENT OR COMMON REPRESENTATIVE; OR ADDRESS FOR CORRESPONDENCE

The following person is [X] agent [ ] common representative

and [X] has been appointed earlier and represents the applicant(s) also for international preliminary examination.

[ ] is hereby appointed and any earlier appointment of (an) agent(s)/common representative is hereby revoked.

[ ] is hereby appointed, specifically for the procedure before the International Preliminary Examining Authority, in addition to the agent(s)/common representative appointed earlier.

Name and address: *(Family name followed by given name; for a legal entity, full official designation. The address must include postal code and name of country.)*

Plougmann, Vingtoft & Partners A/S
Sankt Annæ Plads 11
DK-1250 Copenhagen K
Denmark

Telephone No.: +45 33 63 93 00

Facsimile No.: +45 33 63 96 00

Teleprinter No.: -

[ ] **Address for correspondence:** Mark this check-box where no agent or common representative is/has been appointed and the space above is used instead to indicate a special addr ess to which correspondence should be sent.

Box No. IV BASIS FOR INTERNATIONAL PRELIMINARY EXAMINATION

Statement concerning amendments:*

1. The applicant wishes the international preliminary examination **to start on the basis of:**

[X] the international application as originally filed

the description
[ ] as originally filed
[ ] as amended under Article 34

the claims
[ ] as originally filed
[ ] as amended under Article 19 (together with any accompanying statement)
[ ] as amended under Article 34

the drawings
[ ] as originally filed
[ ] as amended under Article 34

2. [ ] The applicant wishes any amendment to the claims under Article 19 to be considered as reversed.

3. [ ] The applicant wishes the start of the international preliminary examination **to be postponed** until the expiration of 20 months from the priority date unless the International Preliminary Examining Authority receives a copy of any amendments made under Article 19 or a notice from the applicant that he does not wish to make such amendments (Rule 69.1(d)). *(This check-box may be marked only where the time limit under Article 19 has not yet expired.)*

* Where no check-box is marked, international preliminary examination will start on the basis of the international application as originally filed or, where a copy of amendments to the claims under Article 19 and/or amendments of the international application under Article 34 are received by the International Preliminary Examining Authority before it has begun to draw up a written opinion or the international preliminary examination report, as so amended.

Language for the purposes of international preliminary examination: English

[X] which is the language in which the international application was filed.
[ ] which is the language of a translation furnished for the purposes of international search.
[ ] which is the language of publication of the international application.
[ ] which is the language of the translation (to be) furnished for the purposes of international preliminary examination.

Box No. V ELECTION OF STATES

The applicant hereby **elects all eligible States** *(that is, all States which have been designated and which are bound by Chapter II of the PCT)*

excluding the following States which the applicant wishes **not to elect:**

Sheet No. 4

International application No.
PCT/DK00/00354

Box No. VI CHECK LIST

The demand is accompanied by the following elements, in the language referred to in Box No. IV, for the purposes of international preliminary examination:

				For International Preliminary Examining Authority use only	
				received	not received
1. translation of international application	:		sheets	☐	☐
2. amendments under Article 34	:		sheets	☐	☐
3. copy (or, where required, translation) of amendments under Article 19	:		sheets	☐	☐
4. copy (or, where required, translation) of statement under Article 19	:		sheets	☐	☐
5. letter	:	1 ✓	sheets	☐	☐
6. other *(specify)*	:		sheets	☐	☐

The demand is also accompanied by the item(s) marked below:

1. [x] fee calculation sheet
2. [ ] separate signed power of attorney
3. [ ] copy of general power of attorney; reference number, if any:
4. [ ] statement explaining lack of signature
5. [ ] nucleotide and or amino acid sequence listing in computer readable form
6. [ ] other *(specify)*:

Box No. VII SIGNATURE OF APPLICANT, AGENT OR COMMON REPRESENTATIVE

Next to each signature, indicate the name of the person signing and the capacity in which the person signs (if such capacity is not obvious from reading the demand).

26 January 2001
Plougmann, Vingtoft & Partners A/S

Jakob Pade Frederiksen

For International Preliminary Examining Authority use only

1. Date of actual receipt of DEMAND:
2. Adjusted date of receipt of demand due to CORRECTIONS under Rule 60.1(b):
3. [ ] The date of receipt of the demand is AFTER the expiration of 19 months from the priority date and item 4 or 5, below, does not apply. [ ] The applicant has been informed accordingly.
4. [ ] The date of receipt of the demand is WITHIN the period of 19 months from the priority date as extended by virtue of Rule 80.5.
5. [ ] Although the date of receipt of the demand is after the expiration of 19 months from the priority date, the delay in arrival is EXCUSED pursuant to Rule 82.

For International Bureau use only

Demand received from IPEA on:

PATENT COOPERATION TREATY

From the:
INTERNATIONAL PRELIMINARY EXAMINING AUTHORITY

PLOUGMANN VINGTOFT & PARTNERS

16 MRS. 2001

PCT

JJS/KHO

WRITTEN OPINION

(PCT Rule 66)

To:

PLOUGMANN, VINGTOFT & PARTNERS
Sankt Ann Plads 11
DK-1250 Copenhagen K
DANEMARK

Date of mailing *(day/month/year)*	14.03.2001
Applicant's or agent's file reference: 25107 PC 1	**REPLY DUE** **within 3 month(s)** from the above date of mailing

International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/DK00/00354	30/06/2000	21/07/1999

International Patent Classification (IPC) or both national classification and IPC
G06F17/30

Applicant
MINDPASS A/S et al.

1. This written opinion is the **first** drawn up by this International Preliminary Examining Authority.

2. This opinion contains indications relating to the following items:

I	☒	Basis of the opinion
II	☐	Priority
III	☐	Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
IV	☐	Lack of unity of invention
V	☒	Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
VI	☐	Certain document cited
VII	☒	Certain defects in the international application
VIII	☒	Certain observations on the international application

3. The applicant is hereby **invited to reply** to this opinion.

When? See the time limit indicated above. The applicant may, before the expiration of that time limit, request this Authority to grant an extension, see Rule 66.2(d).

How? By submitting a written reply, accompanied, where appropriate, by amendments, according to Rule 66.3. For the form and the language of the amendments, see Rules 66.8 and 66.9.

Also: For an additional opportunity to submit amendments, see Rule 66.4.
For the examiner's obligation to consider amendments and/or arguments, see Rule 66.4 bis.
For an informal communication with the examiner, see Rule 66.6.

If no reply is filed, the international preliminary examination report will be established on the basis of this opinion.

4. The final date by which the international preliminary examination report must be established according to Rule 69.2 is: 21/11/2001.

Name and mailing address of the international preliminary examining authority:
European Patent Office
D-80298 Munich
Tel. +49 89 2399 - 0 Tx: 523656 epmu d
Fax: +49 89 2399 - 4465

Authorized officer / Examiner
Nippl, C

Formalities officer (incl. extension of time limits)
Schall, H
Telephone No. +49 89 2399 2647

I. Basis of the opinion

1. This opinion has been drawn on the basis of (*substitute sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this opinion as "originally filed".*):

Description, pages:

1-48 as originally filed

Claims, No.:

1-28 as originally filed

Drawings, sheets:

1/4-4/4 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:

☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

   (Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

V. Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrialapplicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Claims
Inventive step (IS)	Claims
Industrial applicability (IA)	Claims

2. Citations and explanations
   see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

VIII. Certain observations on the international application

The following observations on the clarity of the claims, description, and drawings or on the question whether the claims are fully supported by the description, are made:
see separate sheet

Re Item V

Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Reference is made to the following documents:

   D1: JIM GRAY ET AL: 'Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals' DATA MINING AND KNOWLEDGE DISCOVERY 1, 1997, pages 29-53, XP002901286

   D2: H.J. LENZ ET AL: 'Summarizability in OLAP and Statistical Data bases' SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 1997. PROCEEDINGS, NINTH INTERNATIONAL CONFERENCE , pages 132-143, XP002901287

   D3: INDERPAL SINGH MUMICK: 'Maintenance of Data Cubes and Summary Tables in a Warehouse' AT&T LABORATORIES, [Online] 1997, XP002901288

2. As far as independent claims 1, 15 and 23 can be understood (see Section VIII below), they seem to relate to the problem of modifying existing dimension hierarchies in a multidimensional data space in order to provide summarizability. Summarizability is an important property in statistical and OLAP applications, because violating this condition can lead to erroneous conclusions and decisions when using aggregates.

2.1 Document D1, which is considered to represent the most relevant state of the art, discloses a new database operator that generalizes aggregations for the N-dimensional data space. Solutions are proposed on how to integrate this operator on the execution and (SQL) language level.
D2 discloses a method for testing the summarizability condition, without proposing a solution on how to establish this condition.
D3 deals with maintenance problems of aggregates in the case of updates.

2.3 D1 only mentions the problem of irregular dimension hierarchies where the usage of pre-aggregates is not possible. A solution to this problem is proposed in claims 1,15 and 23 of the present application by creating new dimension values and modifying the mapping among dimension values accordingly.

Additionally, claim 23 sets out a method on how to keep this procedure transparent to the user by using two sets of dimensions.
This solution is not known nor rendered obvious by the available prior art and is thus considered as involving an inventive step (Article 33(3) PCT).
However, the attention of the Applicant is drawn to the fact that this opinion holds only if the claims are amended in order to overcome the objections with respect to clarity set forth in Section VIII.

Re Item VIII

Certain observations on the international application

1. The application does not meet the requirements of Article 6 PCT, because claims 1-28 are not clear.

1.1. The following terms are vague and indefinite or have no well-recognised meaning. Thus they leave the reader in doubt as to the meaning of the technical features to which they refer, thereby rendering the definition of the subject-matter of said claims unclear and ambiguous:
Claim 1: "(partly) aggregation normalized", "dimension values", "irregularities"
Claim 2: "strict", "covering", "onto"
Claim 3: "into"
Claim 6: "bottom category"
Claim 11: "facts", "multidimensional object"
Claim 19, 23: "aggregate/navigation queries"
Claim 23: "irregular"
In order to overcome this objection, the essential technical features of the above terms should be clearly defined.
The term "aggregation" is especially problematic, since it can be interpreted differently e.g. in the field of object-oriented programming, or even in the patent law literature, where it designates a combination of features which is not patentable because there is no functional relationship between the features (see PCT Guidelines IV-8.3.a). This is similar for the term "dimension".
Different interpretations in said fields can lead to different results when assessing inventive step. Therefore it seems to be essential to include the term "on-line analytical processing" into the wording of the independent claims in order to clarify

the envisaged technical field.
Furthermore it is essential to provide a definition of the terms "aggregation normalise", "onto", "covering" and "strict" so that the meaning is clear from the wording of the claims alone (see PCT Guidelines, III-4.2).

2. Furthermore, there is no well-defined antecedent basis for the phrases "the execution" in claim 3 and 4, "above categories" in claim 5, "according to the method" in claim 19, line 34.
3. It is assumed that claim 7, line 20 should read "top category".
4. Claims 13 and 14 attempt to define the subject-matter in terms of the result to be achieved which merely amounts to a statement of the underlying problem. The technical features necessary for achieving this result should be added.

Re Item VII

Certain defects in the international application

1. Contrary to the requirements of Rule 5.1(a)(ii) PCT, the relevant background art disclosed in document D1 is not mentioned in the description.
2. Furthermore, according to Rule 5.1(a)(iv) PCT, a summary of the drawings should be provided in the description.
3. The description is not consistent with the figures in the sense that erroneous references are used: e.g. page 31, line 1; page 36, line 19; page 37, line 11, as well as on page 38 and 39.
4. The applicant is requested to file amendments by way of replacement pages in the manner stipulated by Rule 66.8(a) PCT. In particular, fair copies of the amendments should be filed preferably in triplicate. Moreover, the applicant's attention is drawn to the fact that, as a consequence of Rule 66.8(a) PCT the examiner is not permitted to carry out any amendments under the PCT procedure, however minor these may be.
5. In order to facilitate the examination of the conformity of the amended application with the requirements of Article 34(2)(b) PCT, the applicant is requested to clearly identify the amendments carried out, no matter whether they concern amendments by addition, replacement or deletion, and to indicate the passages of the application as filed on which these amendments are based (see also Rule 66.8(a) PCT). If the applicant regards it as appropriate these indications could be

submitted in handwritten form on a copy of the relevant parts of the application as filed.

10/031911

PATENT COOPERATION TREATY

PLOUGMANN VINGTOFT & PARTNERS

06 AUG. 2001

From the
INTERNATIONAL PRELIMINARY EXAMINING AUTHORITY

To:

PLOUGMANN, VINGTOFT & PARTNERS
Sankt Ann Plads 11
DK-1250 Copenhagen K
DANEMARK

PCT JJS/HHO

NOTIFICATION OF TRANSMITTAL OF THE INTERNATIONAL PRELIMINARY EXAMINATION REPORT
(PCT Rule 71.1)

Date of mailing *(day/month/year)*	03.08.2001
Applicant's or agent's file reference 25107 PC 1	**IMPORTANT NOTIFICATION**

International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/DK00/00354	30/06/2000	21/07/1999

Applicant
MINDPASS A/S et al.

1. The applicant is hereby notified that this International Preliminary Examining Authority transmits herewith the international preliminary examination report and its annexes, if any, established on the international application.

2. A copy of the report and its annexes, if any, is being transmitted to the International Bureau for communication to all the elected Offices.

3. Where required by any of the elected Offices, the International Bureau will prepare an English translation of the report (but not of any annexes) and will transmit such translation to those Offices.

4. **REMINDER**

The applicant must enter the national phase before each elected Office by performing certain acts (filing translations and paying national fees) within 30 months from the priority date (or later in some Offices) (Article 39(1)) (see also the reminder sent by the International Bureau with Form PCT/IB/301).

Where a translation of the international application must be furnished to an elected Office, that translation must contain a translation of any annexes to the international preliminary examination report. It is the applicant's responsibility to prepare and furnish such translation directly to each elected Office concerned.

For further details on the applicable time limits and requirements of the elected Offices, see Volume II of the PCT Applicant's Guide.

Name and mailing address of the IPEA/	Authorized officer
European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Atienza Vivancos, B Tel.+49 89 2399-7891

Form PCT/IPEA/416 (July 1992)

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference	FOR FURTHER ACTION	See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)
25107 PC 1		
International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/DK00/00354	30/06/2000	21/07/1999
International Patent Classification (IPC) or national classification and IPC G06F17/30		
Applicant MINDPASS A/S et al.		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 5 sheets, including this cover sheet.

   ☒ This report is also accompanied by ANNEXES, i.e. sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

   These annexes consist of a total of 22 sheets.

3. This report contains indications relating to the following items:

   - I ☒ Basis of the report
   - II ☐ Priority
   - III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
   - IV ☐ Lack of unity of invention
   - V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations suporting such statement
   - VI ☐ Certain documents cited
   - VII ☒ Certain defects in the international application
   - VIII ☐ Certain observations on the international application

Date of submission of the demand	Date of completion of this report
29/01/2001	03.08.2001
Name and mailing address of the international preliminary examining authority: European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized officer Nippl, C Telephone No. +49 89 2399 7372

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/DK00/00354

I. Basis of the report

1. With regard to the **elements** of the international application *(Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rules 70.16 and 70.17))*:
Description, pages:

1-48 as originally filed

Claims, No.:

1-28 as originally filed

Drawings, sheets:

1/4-4/4 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:

☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

(Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-28
	No:	Claims	
Inventive step (IS)	Yes:	Claims	1-28
	No:	Claims	
Industrial applicability (IA)	Yes:	Claims	1-28
	No:	Claims	

2. Citations and explanations
see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

Re Item V

Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Reference is made to the following documents:

   D1: JIM GRAY ET AL: 'Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals' DATA MINING AND KNOWLEDGE DISCOVERY 1, 1997, pages 29-53, XP002901286

   D2: H.J. LENZ ET AL: 'Summarizability in OLAP and Statistical Data bases' SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 1997. PROCEEDINGS, NINTH INTERNATIONAL CONFERENCE , pages 132-143, XP002901287

   D3: INDERPAL SINGH MUMICK: 'Maintenance of Data Cubes and Summary Tables in a Warehouse' AT&T LABORATORIES, [Online] 1997, XP002901288

2. Independent claims 1, 15 and 23 relate to the problem of modifying existing dimension hierarchies in a multidimensional data space in order to provide summarisability.
   Summarisability is an important property in statistical and OLAP applications, because violating this condition can lead to erroneous conclusions and decisions when using aggregates.

2.1 Document D1, which is considered to represent the most relevant state of the art, discloses a new database operator that generalizes aggregations for the N- dimensional data space. Solutions are proposed on how to integrate this operator on the execution and (SQL) language level.
D2 discloses a method for testing the summarisability condition, without proposing a solution on how to establish this condition.
D3 deals with maintenance problems of aggregates in the case of updates.

2.3 In document D1 it is mentioned that irregular dimension hierarchies render pre-aggregation impossible but no solution to this problem is provided.

In claims 1,15 and 23 of the present application this problem is solved by creating new dimension values and modifying the mapping among dimension values accordingly. This method enables pre-aggregates also for dimension hierarchies which are not covering, onto and strict.

Additionally, claim 23 sets out a method on how to keep this procedure transparent to the user by using two sets of dimensions.

This solution is not known nor rendered obvious by the available prior art and is thus considered as involving an inventive step (Article 33(3) PCT).

Re Item VII

Certain defects in the international application

1. The description (summary of the invention) is not in conformity with the claims as required by Rule 5.1(a)(iii) PCT.

25107 PC

KMJ
Karin Moelker Jørgen

AMENDED SET OF CLAIMS
PCT/DK00/00354
REPLY OF 12 JUNE 2001 TO WRITTEN OPINION

1. A method for transforming a general on-line analytical processing dimension into an at least partly aggregation normalised dimension, i.e. a dimension with improved summarisability, by means of a computer, the dimension having dimension values organised into categories of dimension values based on a partial ordering, the dimension comprising mappings of links between dimension values, the method comprising the steps of

retrieve the mapping from data storage means associated with the computer,

analysing the mapping to determine irregularities of the dimension , i.e. features rendering the dimension non-summerisable, by means of analysing means executed by the computer,

creating new dimension values of the dimension and modifying the mapping between dimensional values of the dimension according to the analysis, whereby the dimension is at least partly aggregation normalised, and

saving the new dimension values and the modified mappings in data storage means of the computer.

2. A method according to claim 1, wherein the step of creating new dimensional values and modifying the mapping comprises the steps of

examine whether the dimension is covering, i.e. that only immediate parent and child values can be related, as well as onto, i.e. that all paths in the hierarchy have equal lengths, and in case it is,

executing a make-strict procedure for making the dimension aggregation strict, i.e. that each child in a hierarchy has only one parent, thereby making the non-strict dimension aggregation normalised, i.e. summarisable.

3. A method according to claim 1 or 2, wherein the step of creating new dimensional values and modifying the mapping comprises the steps of

examine whether the dimension is covering, and in case it is,

executing a make-onto procedure for making the dimension onto, thereby at least partly making an non-onto dimension aggregation normalised.

4. A method according to any of claims 1-3, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the dimension covering, thereby at least partly making a non-covering dimension aggregation normalised.

5. A method according to any of claims 2-4, wherein the make-strict procedure comprises the steps of, starting from the bottom category and successively proceeding towards the top category,

identifying combinations of dimensional values of the same category for each of which combination at least one dimension value of a category below said category is linked to each of the dimension values of the combination,

creating one or more new dimensional values each representing one of the identified combinations of dimensional values and creating links from the new dimensional values to dimensional values of above categories in accordance with existing links from each of the dimensional values represented by the new dimensional value, and

identifying dimension values being linked to identified combinations of dimensional values of the same category and replacing the links with links to new dimensional values representing said combinations of dimensional values.

6. A method according to any of claims 2-5, wherein the make-strict procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category being a direct predecessor of the child category of which category at least one dimension value of the child category is linked to a dimension value of, setting said category as the parent category and performing the steps of:

(iia) ending the make-strict procedure for the parent category in case the parent category is the top category of the dimension,

(iib) ending the make-strict procedure for the parent category in case no dimension value of the parent category is linked to a dimension value of a higher category,

(iic) creating a new fused category in the dimension immediately below the parent category in case at least one of the dimension values of the child category is linked to more than one dimension value of the parent category,

(iid) for each dimensional value of the child category, performing the steps of: creating a new dimension value of the new fused category representing the one or more values of the parent category to which the dimensional value of the child category is linked and creating links from said new dimension value to said values in the parent category, the creation of the new dimension value being conditioned that no dimension value of the new fused category already exists having exactly such link(s), and

for each category being a direct predecessor of the parent category of which category at least one dimension value of the parent category is linked to a dimension value of, setting said category as a grandparent category and creating links from the new dimension value to the one or more dimension values of the grandparent category to which said one or more dimensional values of the parent category are linked,

(iie) removing the links from the parent category to the one or more grandparent categories, whereby the grandparent categories no longer are direct predecessors of the parent category,

(iif) creating links from each dimensional value of the child category to the dimension value of the new fused category having the same links to the dimension values of the parent category whereby the new fused category becomes a direct predecessor of the child category, and removing the links from the dimension values of the child category to the parent category, whereby the parent category no longer is a direct predecessor of the child category,

and

(iig) setting the new fused category as the child category and returning to step (ii).

7. A method according to any of claims 3-6, wherein the make-onto procedure comprises the steps of, starting from the top category and successively proceeding towards the bottom category,

creating, for each dimension value of each category above the bottom category not being linked to any dimensional value of the category immediately below, a new dimension value in the category immediately below and creating a link between said new dimension value and said dimension value of the category in question.

8. A method according to any of claims 3-7, wherein the make-onto procedure comprises the successive steps of

(i) setting the top category of the dimension as the parent category,

(ii) for each category immediately below the parent category and having dimension values being linked to dimension values of the parent category, setting said category as the child category and perform the steps of

(iia) creating, for each dimension value of the parent category not being linked to any dimensional value of the child category, a new dimension value in the child category and creating a link between said new dimension value and said dimension value of the parent category,

(iib) setting the child category as parent category,

(iic) ending the make-onto procedure in case the parent category is the bottom category of the dimension, else returning to step (ii) of the make-onto procedure.

9. A method according to any of claims 4-8, wherein the make-covering procedure comprises the successive steps of

identifying links between dimension values of two categories having at least one intermediate category there between,

creating a new dimension value in each of said intermediate categories for each of those links for which no paths of links exists going only through immediate child-parent links from lower to higher categories and including a link to a dimension value of the intermediate category, and

replacing those links with links between the dimension values of those links and the new dimension values.

10. A method according to any of claims 4-9, wherein the make-covering procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category immediately above the child category for which at least one link between a dimension value of said category and a dimension value of the child category exists, setting the category as the parent category and perform the steps of:

(iia) ending the make-covering procedure for the parent category in case the parent category is the top category of the dimension:,

(iib) for each higher category being a direct predecessor category of the child category and being higher in the hierarchy than the parent category, performing the steps of

(iiba) identifying sets of dimension values of the higher category and dimension values of the child category for which sets

a link exists, and

no paths of links going only from lower to higher categories and including a link to a dimension value of the parent category exists, and

(iibb) creating for each identified set of dimension values a new dimension value in the parent category, creating links between each of the dimension values of the set and the new dimension value, and removing the link between the two dimension values of the identified set, whereby the higher category no longer is a predecessor of the child category,

(iic) setting the parent category as the child category and returning to step (ii).

11. A method for by means of a computer to at least partly aggregation normalise a multidimensional object including a set of facts comprising a plurality of facts mapped on a plurality of dimensions having dimension values organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension, by means of applying the method of any of claims 1-10 to at least one of the dimensions of the multidimensional object.

12. A method according to claim 11, wherein the multidimensional object comprises a plurality of facts and the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, the facts constituting the bottom layer of each of the dimensions of the multidimensional object.

13. A method according to claim 11 or 12, comprising the steps of

selecting a subset of categories of the one or more dimension to be aggregation normalised, and

performing an aggregation normalisation of the selected subset,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised.

14. A method according to any of claims 11-13, comprising the steps of

selecting specific aggregation functions to be performed on the multidimensional object, and

selecting by means of the computer normalisation steps to be performed based on the selection of specific aggregation functions to be performed,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised. .

15. A method for by means of a computer to at least partly aggregation normalise a general on-line analytical processing multidimensional object including a set of facts comprising a plurality of facts mapped on an aggregation normalised plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension,

the method comprising the steps of

retrieve the mapping from data storage means associated with the computer, including the mapping of the plurality of facts on the multidimensional object into the mapping of the multidimensional object so that the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, and the facts constitutes the bottom layer of each of the dimensions of the multidimensional object,

analysing the mapping of the multidimensional object to determine irregularities of the dimensions by means of analysing means executed by the computer,

creating new dimension values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is at least partly aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

16. A method according to claim 15, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-strict procedure for making the multidimensional object aggregation strict, thereby making the non-strict multidimensional object aggregation normalised, the make-strict procedure being executed on the condition that the multidimensional object is covering prior to the execution.

17. A method according to claim 15 or 16, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the multidimensional object covering, thereby at least partly making the non-covering multidimensional object aggregation normalised.

18. A method according to any of claims 15-17, wherein the method comprises the initial step of making each of the plurality of dimensions aggregation normalised by means of the method according to any of claims 1-10.

19. A method according to any of claims 11-18, wherein the created new dimensional values are marked as such, a pre-aggregation is performed on a multidimensional object being normalised by means of the computer and the method further comprises the step of producing a reply to a query made to the system and concerning the multidimensional object, aggregate queries, exploring the dimension hierarchies, as well as navigation queries, that summarise the data at various levels of detail, in which reply the existence of the created new dimensional values is transparent.

20. A method according to any of claims 11-19, further comprising the steps of implementing, into the aggregation normalised multidimensional object, of new facts including mapping of the facts onto the dimension of new dimension values of the dimensions, or of new mapping between some of the dimension values, by which implementation irregularities of the multidimensional object is introduced,

analysing the introduced irregularities of the dimensions of the multidimensional object,

creating new dimensional values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

21. A computer system comprising at least one general purpose computer having data storage means associated therewith on which data storage means is stored a computer programme product suitable for adapting the computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20, the computer system comprising means for retrieving the computer programme product and perform accordingly.

22. A computer programme product suitable for adapting a general purpose computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

23. A computer system for on-line analytical processing having data storage means associated therewith on which a multidimensional object is stored, the multidimensional object including

a set of facts comprising a plurality of facts,

a first plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a first mapping of links between dimension values within each dimension of the first plurality of dimensions as well as links between the facts and the dimensions of the first plurality of dimensions, at least one of the dimensions of the first plurality of dimensions being irregular, and

a second plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a second mapping of links between dimension values within each dimension of the second plurality of dimensions as well as links between the facts and the dimensions of the second plurality of dimensions, each of the second plurality of dimensions being aggregation normalised,

the computer system comprising a query handler component being adapted for producing replies to queries made to the computer system and concerning the multidimensional object, the replies to navigation queries being based on the first set of dimensions and the replies to aggregate queries being based on the second set of dimensions.

24. A computer system according to claim 23, wherein a set of pre-aggregation data relating to the second plurality of dimensions is further stored within the data storage means and the replies to aggregate queries furthermore are based on the set of pre-aggregation data.

25. A computer system according to claim 23 or 24, wherein the query handler component is adapted for producing replies to aggregate queries in which replies the existence of the second plurality of dimensions is transparent.

26. A computer system according to claim 25, wherein the query handler component is adapted for transforming aggregate queries made to the first plurality of dimensions into queries for the second set of dimensions and transforming replies based on the second set of dimensions into replies as based on the first set of dimensions, thus making the existence of the second plurality of dimensions transparent in the produced reply.

27. A computer system according to claim 26, wherein the multidimensional object is stored within the data storage means of the computer system in tables organised as a combination of star schemes for the part of the multidimensional object containing only strict mappings, and additional tables containing the non-strict part of the mappings, the query handler component makes use of said tables in transforming queries and replies.

28. A computer system according to any of claims 23-27 further comprising means adapted for performing an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

combinations grows rapidly when the number of dimensions increase, while the sparseness of the multidimensional space decreases in higher dimension levels, meaning that aggregates at higher levels take up nearly as much space as lower-level aggregates. In some commercial applications, full pre-aggregation takes up as much as 200 times the space of the raw data [21]. Another problem with full pre-aggregation is that it takes too long to update the materialised aggregates when base data changes.

With the goal of avoiding data explosion, research has focused on how to select the best subset of aggregation levels given space constraints [1, 9, 11, 26, 28, 32] or maintenance time constraints [10], or the best combination of aggregate data and indices [8]. This approach is commonly referred to as *practical* (or partial or semi-eager [5, 11, 29]) pre-aggregation. Commercial OLAP systems now also exist that employ practical pre-aggregation, e.g., Microsoft Decision Support Services (Plato) [18] and Informix MetaCube [13].

< page 2a

The premise underlying the applicability of practical pre-aggregation is that lower-level aggregates can be *re-used* to compute higher-level aggregates, known as summarisability [16]. Summarisability occurs when the mappings in the dimension hierarchies are *onto* (all paths in the hierarchy have equal lengths), *covering* (only immediate parent and child values can be related), and *strict* (each child in a hierarchy has only one parent); and when also the relationships between facts and dimensions are many-to-one and facts are always mapped to the lowest levels in the dimensions [16]. However, the data encountered in many real-world applications fail to comply with this rigid regime. This motivates the search for techniques that allow practical pre-aggregation to be used for a wider range of applications, the focus of the present invention.

Description of the Invention

Motivated by the increasing use of OLAP systems in many different applications, including in business and health care, the present invention provides transformation techniques for multidimensional databases that leverage the existing, performance-enhancing technique, known as practical, or partial or semi-eager, pre-aggregation, by making this technique relevant to a much wider range of real-world applications.

Current pre-aggregation techniques assume that the dimensional structures are *summarisable*. Specifically, the mappings in dimension hierarchies must be *onto*,

2a

A new database operator that generalises aggregations for the N-dimensional data space is disclosed by Jim Gray et al. *"Data Cube: A Relational Aggregation Operator Generalizing group-By, Cross-Tab and Sub-Totals", Data Mining and Knowledge Discovery 1, 1997*, and solutions are proposed on how to integrate this operator on the execution and SQL-language level. It is mentioned that irregular dimension hierarchies renders the pre-aggregation impossible but no solution to this problem is provided.

Detailed description of the Invention

We now proceed to describe the invention in detail. The next section presents a real-world clinical case study that exemplifies the non-summarisable properties of real-world applications. The following section proceeds to define the aspects of a multidimensional data model necessary for describing the new techniques, and defines also important properties related to summarisability. Algorithms are presented for transforming dimension hierarchies to achieve summarisability, then apply the algorithms to fix non-summarisable relationships between facts and dimensions. It is also demonstrated how the techniques may be integrated into current systems, transparently to the user and how to modify the algorithms to accommodate incremental computation.

< page 13a

Brief description of figures

The detailed description of the invention is accompanied by a set of figures of which

Fig. 1 is an ER diagram illustrating the underlying data of the case study,

Fig. 2 illustrates the dimension types of the case study,

Fig. 3 illustrates on the left the transformation performed on the hierarchy by the Make Covering algorithm, and on the right the transformation performed on the hierarchy by the Make Onto algorithm,

Fig. 4 illustrates the transformation performed on the hierarchy by the Make Strict algorithm,

Fig.5 illustrates on the left another example of the transformation performed by the Make Onto algorithm and on the right the transformation performed by the Make Strict algorithm executed thereafter,

Fig. 6 shows on the left the architecture of a standard OLAP system and on the right the architecture of the present invention, and

Fig. 7 shows the implementation of the system architecture,

properties, so applying each once is sufficient.

In general, the algorithms take as input a set of tables R_{C_1,C_2} that specifies the mapping from dimension values in category C_1 to values in category C_2. The input needs not contain all pairs of ancestors and descendants—only direct parent-child relationships are required. If there are non-covering mappings in the hierarchy, we have categories C, P, H such that $\{P, H\} \subseteq Pred(C)$ and $Type(P) < Type(H)$. In this case, the input must also contain $R_{P,H}$ tables that map P values to H values.

The algorithms are expressed using recursion. They could also easily be expressed using iteration instead.

Non-Covering Hierarchies

The first algorithm renders all mappings in a dimension hierarchy covering w.r.t. any category. When a dimension value is mapped *directly* to another value in a category higher than the one immediately above it in the hierarchy, a new intermediate value is inserted into the category immediately above, and the two original dimension values are linked to this new value, rather than to each other.

Example 6 In the hierarchy for the Residence dimension, two links go from Address directly to County. The address "123 Rural Road" (52) is in "Melbourne County" (31), but not in a city, and the address "1 Sandy Dunes" (53) is in "Outback County" (32), which does *not* have any cities at all. The algorithm inserts two new dimension values in the City category, **C31** and **C32**, which represent Melbourne and Outback county, respectively, and links them to their respective counties. The addresses "123 Rural Road" and "1 Sandy Dunes" are then linked to **C31** and **C32**, respectively. This occurs in the first call of procedure MakeCovering (on the Address category; the procedure is given below). When MakeCovering is called recursively on the City, County, and T categories, nothing happens, as all mappings are already covering. The transformation is illustrated graphically in Figure 3. The dotted lines show the "problematic" links, and the bold-face values and thick lines show the new dimension values and links.

In the algorithm, C is a *child* category, P is an immediate *parent* category, H is a "*higher*" category, L are the non-covering *links* from C to H, and N are the "higher" dimension values in L. The $\bowtie$ operator denotes natural join. The algorithm works as follows. Given the argument category C (initially the bottom category) in line (1), the algorithms goes through all C's

< to the left >

Non-Onto Hierarchies

The second algorithm renders all mappings in hierarchies onto, i.e., all dimension values in non-bottom categories have children. This is ensured by inserting placeholder values in lower categories to represent the childless values. These new values are marked with the original values, making it possible to map facts to the new placeholder values instead of to the original values. This makes it possible to only map facts to the bottom category.

Example 7 In the Diagnosis dimension, the "Lung cancer" diagnosis family (ID = 14) has no children. When the algorithm reaches the Diagnosis Family category, it inserts a placeholder value (**L14**) into the Low-level Diagnosis category, representing the "Lung cancer" diagnosis, and links it to the original value. Facts mapped to the "Lung cancer" value may then instead be mapped to the new placeholder value, ensuring that facts are mapped only to the Low-level Diagnosis Category. A graphical illustration of the transformation is seen ~~in Figure 4~~ to the right in Figure 3. The bold-faced **L14** value is the new value inserted; and the thick line between 14 and **L14** is the new link inserted.

In the algorithm below, P is a *parent* category, C is a *child* category, and N holds the parent values with *no* children. The algorithm works as follows. Given a category P (initially the $\top$ category) in line (1), the algorithm goes through all categories C that are (immediate) descendants of P (2). For each C, line (4) finds the values N in P that have *no* children in C, by "subtracting" the values *with* children in C from the values in P. For each "childless" value in N, lines (5) and (6), respectively, insert into C a placeholder value marked with the parent value, and links the new value to the original. MakeOnto is then called recursively on C (7). The algorithms terminates when it reaches the $\bot$ category, which has no descendants.

```
(1)  procedure MakeOnto(P)
(2)      for each C ∈ Desc(P) do
(3)      begin
(4)          N ← P \ Π_P(R_{C,P})
(5)          C ← C ∪ {Mark(p) | p ∈ N}
(6)          R_{C,P} ← R_{C,P} ∪ {(Mark(p), p) | p ∈ N}
(7)          MakeOnto(C)
(8)      end
(9)  end
```

ure 4. Because of the non-strictness in the mapping from Low-level Diagnosis to Diagnosis Family, and from Diagnosis Family to Diagnosis Group, two new category types and the corresponding categories are introduced. The third picture indicates the argument to the algorithm; and, in addition, its dotted lines indicate the links deleted by the algorithm. The fourth picture gives the result of applying the algorithm; here, the bold-face values and thick lines indicate the values and links inserted by the algorithm.

In the first call of the algorithm the three Low-level Diagnosis values—"(low-level) Lung cancer" (L14); "Insulin dependent diabetes during pregnancy" (5); and "Non insulin dependent diabetes during pregnancy" (6)—are linked to the three new fused values—"(low-level) Lung cancer" (14); "Diabetes during pregnancy, Insulin dependent diabetes" (4, 9); and "Diabetes during pregnancy, Non insulin dependent diabetes" (4, 10)—and these are in turn linked to "Lung Cancer" (14); "Diabetes during pregnancy" (4); "Insulin dependent diabetes" (9); and "Non insulin dependent diabetes" (10). The these latter four values in the Diagnosis Family category are un-linked from their parents, as the Diagnosis Family category is "unsafe."

When called recursively on the Set-of Diagnosis Family category, the algorithm creates the new fused values "Cancer" (13) and "Diabetes, Other pregnancy related diseases" (11, 12) in the Set-of Diagnosis Group category. These new values are linked to the values "Cancer" (13), "Diabetes" (11), and "Other pregnancy related diseases" (12) in the Diagnosis Group category, and to the T value; and the values in the Diagnosis Group category are un-linked from their parents. Note the importance of having a T value: the values not linked to T are exactly the unsafe values, for which aggregate results should not be re-used.

The algorithm assumes that all paths in the dimension hierarchy have equal length, i.e., all direct links are from children to their immediate parents. This is ensured by the MakeCovering and MakeOnto algorithms. In the algorithm below, C is a *child* category, P is a *parent* category, G is a *grandparent* category, N is the *new* category introduced to hold the "fused" values, and $\bowtie$ denotes natural join.

The algorithm takes a category C (initially the $\bot$ category) as input. I then goes through the set of immediate parent categories P of C (line (2)). Line (4) tests if there is non-strictness in the mapping from C to P *and* if P has any parents (4). If this test fails, there is no problem as aggregate results for P can either be safely re-used or are guaranteed not be re-used; and the algorithm in then invoked recursively, in line (20).

If the test succeeds, the algorithm creates a new fused category. First, a new, empty category N with domain 2^P is created in line (6). The values inserted into this category represent *sets* of values of P. For example, the value $\top$, $2^{\top}$ represents the set consisting of

F, which is now the bottom of the lattice. The algorithm makes the mappings covering w.r.t. the facts by inserting new marked values, representing the parent values, in the intermediate categories, and by linking the facts to the new values instead of the parent values. As in the section "Non-Covering Hierarchies," the marked values keep information about their original values, so that when new fact-dimension mappings are added, the links that are supposed to go *directly* to the original parent values now instead can be set to go to the marked value in the $\perp$ category.

Example 9 In the case study, the mapping between Patients and Diagnoses is of mixed granularity: "John Doe" (1) and "Jane Doe" are both mapped to the Diagnosis Family, "Insulin dependent diabetes" (9), "Jane Doe" is additionally mapped to the Low-level Diagnosis, "Insulin dependent diabetes during pregnancy" (5), and "Jim Doe" is mapped to "Diabetes" (11), a Diagnosis Group.

In the first call of the algorithm, two new Low-level Diagnoses are inserted: "L9," representing "Insulin dependent diabetes," and "L11," representing "Diabetes"; and the facts are mapped to these instead of the original values. In the recursive call on Low-level Diagnosis, an "F11" value representing "Diabetes" at the Diagnosis Family level is inserted between "Diabetes" and value "L11."

The transformations are illustrated in Figures ~~6 and 7~~ 5, where dotted lines indicate links that are deleted by the algorithm and bold-face value and thick lines indicate dimension values and links inserted by the algorithm.

Many-To-Many Relationships

The second case occurs when relationships between facts and dimension values are many-to-many. This means that the hierarchy, with the facts as the bottom category, is non-strict, leading to possible double-counting of facts. It is enough to make the hierarchy partly strict, as described in the section "Non-Strict Hierarchies." The MakeStrict algorithm is initially called on F, which is now the bottom of the hierarchy lattice. Because the MakeCovering algorithm has already been applied, all paths from facts to the T value have equal length, as required by the MakeStrict algorithm.

Some dimension values have no facts mapped to them, leading to an interesting side effect of the algorithm. When the algorithm fuses values and places the fused values in-between the original values, it also deletes the child-to-parent and parent-to-grandparent links. The

fact-less dimension values are then left disconnected from the rest of the hierarchy, with no links to other values.

These fact-less dimension values do not contribute to any aggregate computations and are thus superfluous. To minimise the dimensions, an "Delete-unconnected" algorithm that deletes the fact-less dimension values by traversing the hierarchy starting at the facts is invoked in a postprocessing step. For a hierarchy of height k, this can be done in time $O(kn \log n)$, where n is the size of the mapping between facts and dimensions. Thus, the overall computational complexity is not altered.

Example 10 The relationship between patients and diagnoses is many-to-many. In Example 9, the MO was transformed so that all mappings were covering, as seen in Figure 5, algorithm MakeStrict is applied to this MO. The final result of the application of the algorithm is seen in Figure 5. Values in italics, e.g., *L14*, and dotted lines indicate deleted values and links. Bold-face values and thick lines denote values and links inserted by the algorithm.

Three new categories are introduced: "Set-of Low-level Diagnosis," "Set-of Diagnosis Family," and "Set-of Diagnosis Group," as non-strictness occurs at all levels. Fused values are inserted into these fused categories. For example, values "(low-level) Lung Cancer" (**L14**), "Insulin dependent diabetes during pregnancy, (low-level) Insulin dependent diabetes" (**5, L9**), and "(low-level) Insulin dependent diabetes" (**L9**) are inserted into the "Set-of Low-level Diagnosis" category; and the original values are linked to the new values.

Values "(low-level) Lung cancer" (**L14**), "Lung cancer" (**14**), "Cancer" (**13**), "Non-insulin dependent diabetes during pregnancy" (**6**), and "Non insulin dependent diabetes" (**10**) do not characterise any facts and are deleted by "Delete-unconnected."

Architectural Context

Overview

The overall idea presented in this paper is to take un-normalised MOs and transform them into normalised MOs that are well supported by the practical pre-aggregation techniques available in current OLAP systems. Queries are then evaluated on the transformed MOs. However, we still want the users to see only the original MOs, as they reflect the users' understanding of the domain. This prompts the need for means of handling both the original and the transformed

MOs. This section explores this coexistence.

A current trend in commercial OLAP technology is the separation of the front-end presentation layer from the back-end database server. Modern OLAP applications consist of an OLAP client that handles the user interface and an OLAP server that manages the data and processes queries. The client communicates with the server using a standardised application programming interface (API), e.g., Microsoft's OLE DB for OLAP [17] or the OLAP Council's MDAPI [20]. The architecture of such a system is given to the left in Figure 8.

This separation of client and server facilitates our desire to have the user see the original MO while queries are evaluated against the transformed MO. Studies have shown that queries on a data warehouse consist of 80% *navigational* queries that explore the dimension hierarchies and 20% *aggregation* queries that summarise the data at various levels of detail [14]. Examples of navigational and aggregation queries are "Show me the Low-Level Diagnoses contained in the Insulin-Dependent Diabetes Diagnosis Family" and "Show me the count of patients, grouped by Diagnosis Family," respectively. The navigational queries must be performed on the *original* MO, while the aggregation queries must be performed on the *transformed* MO. This is achieved by introducing an extra "Query Handler" component between the client and the server. The OLAP client sends a query to the query handler, the primary task of which is to determine whether the query is a navigational query (internal to a dimension) or an aggregation query (involving the facts). Navigational queries are passed to one OLAP server that handles the original (navigational) data, while aggregation queries are passed to another OLAP server that manages the transformed (aggregation) data. This extended system architecture is seen to the right in Figure 8. 6

The OLAP server for navigation data needs to support dimension hierarchies which have non-summarisable properties, a requirement not yet supported by many commercial systems today. However, relational OLAP systems using snow-flake schemas [14] are able to support this type of hierarchies, as are some other OLAP systems, e.g., Hyperion (Arbor) Essbase [12]. If the OLAP system available does not have sufficiently flexible hierarchy support, one solution is to build a special-purpose OLAP server that conforms to the given API. This task is not as daunting as it may seem at first because only *navigational* queries need to be supported, meaning that multidimensional queries can be translated into simple SQL "lookup" queries.

We note that the only data needed to answer navigational queries is the hierarchy definitions. Thus, we only need to store the fact data (facts and fact-dimension relations, in our model) once, in the aggregational data, meaning that the overall storage requirement is only slightly larger than storing just the aggregational data. Navigational queries are evaluated on

the original hierarchy definitions and do not need to be re-written by the query handler.

As described in the section "Dimension Transformation Techniques," aggregation queries need to be re-written slightly by adding an extra HAVING clause condition to exclude results for the new values inserted by the transformation algorithms. This can easily be done automatically by the query handler, giving total transparency for the user. Even though the added HAVING clause conditions are only necessary for the covering and onto transformations, they can also be applied to hierarchies transformed to achieve strictness; this has no effect, but simplifies the query rewriting. The new values can also be filtered out using a modified WHERE clause, by performing an inner join with a table containing only the original values, or by using nested SELECT statements as described in the next section.

Concrete Implementation

We now show how the abstract architecture described above can be implemented using standard relational database technology.

The transparency is achieved by working with two versions of each user-specified hierarchy and by using a query rewrite mechanism. This is described in detail later in this section. The overall system architecture is seen in Figure 8. 7

DiagID	Lowlevel	Family	Group
5	Insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Diabetes
5	Insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Pregnancy related
5	Insulin dependent diabetes during pregnancy	Insulin dependent diabetes	Diabetes
6	Non insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Diabetes
6	Non insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Pregnancy related
6	Non insulin dependent diabetes during pregnancy	Non insulin dependent diabetes	Diabetes
100	!Lowlevel!Lung Cancer	Lung cancer	Cancer

Table 2: DDiagnosis Dimension Table

The ROLAP client tool, in this case the ROLAP tool Synchrony, which originated from Kimball's Startracker tool [14], makes SQL requests to the ROLAP database, in this case the Oracle8 RDBMS, using the ODBC standard. We have implemented a special, query-transforming ODBC driver (QTOD) that, based on case-specific metadata, transforms the SQL requests into requests that hide the transformations from the users, returning the query results that the user would expect based on the original hierarchies. A transformed request is submitted to the OLAP DB using an RDBMS-specific ODBC driver. The QTOD component is common to all RDBMSs, so Oracle8 may be replaced by another RDBMS such as IBM DB2, Informix, or MS SQL Server. Another ROLAP tool may also be used, making the solution

```
SELECT DISTINCT Lowlevel
FROM DDiagnosis
WHERE Lowlevel NOT LIKE '!%'
```

Lowlevel
Insulin dependent diabetes during pregnancy
Non insulin dependent diabetes during pregnancy

Table 3: Navigational Query Result

Due to the use of DISTINCT as a quantifier, duplicates are not returned. The NOT LIKE predicate removes the placeholder values inserted into the hierarchy to balance it, which in this case is the value "!Lowlevel!Lung Cancer." As desired, the result is unaffected by the translations.

For *aggregation queries*, it is also possible to achieve transformation transparency, although this is more difficult. For dimensions with non-strictness, a special dimension table is introduced that holds only the part of the normalised hierarchy that does *not* contain non-strictness. In the normalised hierarchy to the right in Figure 5.4 this part is the Low-level Diagnosis category and the two special categories introduced by the normalisation process to hold *sets of diagnosis families* and *sets of diagnosis groups*, respectively. This part of the hierarchy is implemented in the Diagnosis dimension table seen in Table 4.

DiagID	Lowlevel	Family	Group
1000020	!Low-level Diagnosis!Lung cancer	14	13
5	Insulin dependent diabetes during pregnancy	4,9	11,12
6	Non insulin dependent diabetes during pregnancy	4,10	11,12

Diagnosis

Group	SGroup
Cancer	13
Diabetes	11,12
Pregnancy Related	11,12

SGroup

Table 4: Dimension and Group Tables for Aggregation

The "Lowlevel" column contains the normal textual diagnosis description, whereas the

< the left part of >

special "Family" and "Group" columns contain comma-separated ordered lists of the IDs of the sets of values that are represented by the column values. For example, value "4,9" represents the set {4,9}.

We need to capture the remaining part of the hierarchy, which consists of non-strict mappings from a "set-of-X" category to the "X" category, e.g., the mapping of the "set-of-Diagnosis Group" category to the "Diagnosis Group" category to the right in Figure 8, which maps {13} to 13 (Cancer) and {11,12} to 11 (Diabetes) and 12 (Pregnancy Related). This is done by introducing a special table for each such mapping, named by the category prefixed with an "S" (for Set-of). For example, for the Diagnosis Group category, table "SGroup" in Table 4 maps sets of diagnosis groups to the individual diagnosis groups in the sets. The "Group" column represents the diagnosis group, while the "SGroup" column represents the associated set of diagnosis groups.

With these tables available, it is possible to obtain transformation transparency for aggregation queries. A ROLAP aggregation query has the format of the query below that computes the number of patients per diagnosis group.

```
SELECT Diagnosis.Group, SUM(Patient.Count)
FROM Diagnosis,Patient
WHERE Diagnosis.DiagID=Patient.DiagID
GROUP BY Diagnosis.Group
```

This is transformed into the more complex query given next.

```
SELECT SGroup.Group, SUM(QQQQQQ.Count)
FROM Sgroup,
  (SELECT Diagnosis.Group,
          SUM(Patient.Count) AS Count
   FROM Diagnosis,Patient
   WHERE Diagnosis.DiagID=Patient.DiagID
   GROUP BY Diagnosis.Group) QQQQQQ
WHERE QQQQQQ.Group=SGroup.SGroup AND
      SGroup.SGroup NOT LIKE '!%'
GROUP BY SGroup.Sgroup
```

The transformed aggregation query has two parts. The nested table expression computes the number of patients per *set of diagnosis group*, making this available via correlation name QQQQQQ. This part of the hierarchy is a balanced tree, so the RDBMS can safely use pre-aggregated data for optimising the query performance. The result of the nested table

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 February 2001 (01.02.2001)

PCT

(10) International Publication Number
WO 01/08041 A1

(51) International Patent Classification[7]: G06F 17/30

(21) International Application Number: PCT/DK00/00354

(22) International Filing Date: 30 June 2000 (30.06.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
PA 1999 01045 21 July 1999 (21.07.1999) DK

(71) Applicant *(for all designated States except US)*: MIND-PASS A/S [DK/DK]; Vardevej 1, DK-9220 Aalborg Ø (DK).

(71) Applicants and
(72) Inventors: PEDERSEN, Torben, Bach [DK/DK]; Trøjborgvej 32, 2. th., DK-8200 Århus N (DK). JENSEN, Christian, S. [DK/DK]; Fr. Bajers Vej 70, DK-9220 Aalborg Ø (DK). DYRESON, Curtis, E. [US/AU]; 6/180 Ron Penhaligon Drive, Robina, Queensland 4226 (AU).

(74) Agent: PLOUGMANN, VINGTOFT & PARTNERS A/S; Sankt Annæ Plads 11, DK-1250 Copenhagen K (DK).

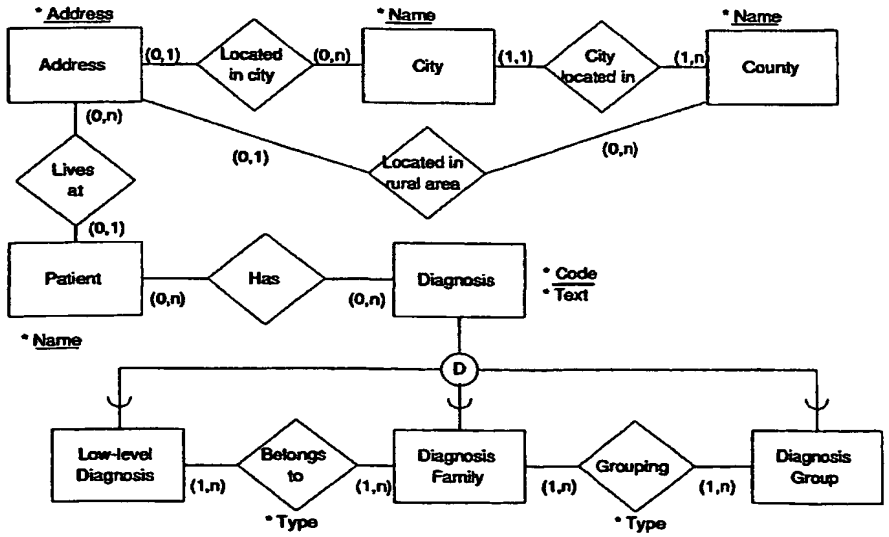
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AT (utility model), AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, CZ (utility model), DE, DE (utility model), DK, DK (utility model), DM, DZ, EE, EE (utility model), ES, FI, FI (utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KR (utility model), KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (utility model), SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

[Continued on next page]

(54) Title: METHOD AND SYSTEMS FOR MAKING OLAP HIERARCHIES SUMMARISABLE



(57) Abstract: A method, a computer system and a computer programme product for a computer system for transforming general On-line Analytical Processing (OLAP) hierarchies into summarisable hierarchies whereby pre-aggregation is disclosed, by which fast query response times for aggregation queries without excessive storage use is made possible even when the hierarchies originally are irregular. Pre-aggregation is essential for ensuring adequate response time during data analysis. Most OLAP systems adopt the *practical pre-aggregation* approach, as opposed to *full pre-aggregation*, of materialising only select combinations of aggregates and then re-use these for efficiently computing other aggregates. However, this re-use of aggregates is contingent on the dimension hierarchies and the relationships between facts and dimensions satisfying stringent constraints. The present invention significantly extends the scope of practical pre-aggregation by transforming irregulare dimension hierarchies and fact-dimension relationships into well-behaved structures that enable practical pre-aggregation.

WO 01/08041 A1

— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

INTERNATIONAL SEARCH REPORT

Intern. Application No
PCT/DK 00/00354

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JIM GRAY ET AL: "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals" DATA MINING AND KNOWLEDGE DISCOVERY 1, 1997, pages 29-53, XP002901286 page 40	1,11-15, 18,20-22
A	---	2-10,16, 17,19, 23-28
A	H.J. LENZ ET AL: "Summarizability in OLAP and Statistical Data bases" SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 1997. PROCEEDINGS, NINTH INTERNATIONAL CONFERENCE , pages 132-143, XP002901287 the whole document	1-28

	-/--	

[X] Further documents are listed in the continuation of box C.

[ ] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 October 2000	24. 11. 2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Oskar Pihlgren

Form PCT/ISA/210 (second sheet) (July 1992)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	INDERPAL SINGH MUMICK: "Maintenance of Data Cubes and Summary Tables in a Warehouse" AT&T LABORATORIES, [Online] 1997, XP002901288 Retrieved from the Internet: <URL:http://citeseer.nj.nec.com/did/38362; > [retrieved on 2000-10-06] the whole document -----	1-28

AMENDED SET OF CLAIMS WITH AMENDMENTS INDICATED
PCT/DK00/00354
REPLY OF 12 JUNE 2001 TO WRITTEN OPINION

---

1. A method for transforming a general on-line analytical processing dimension into an ~~for~~ at least partly aggregation normalised dimension, i.e. a dimension with improved summarisability, by means of a computer, ~~to at least partly aggregation normalise a~~ the dimension having dimension values organised into categories of dimension values based on a partial ordering, the dimension comprising mappings of links between dimension values, the method comprising the steps of

retrieve the mapping from data storage means associated with the computer,

analysing the mapping to determine irregularities of the dimension , i.e. features rendering the dimension non-summerisable, by means of analysing means executed by the computer,

creating new dimension values of the dimension and modifying the mapping between dimensional values of the dimension according to the analysis, whereby the dimension is at least partly aggregation normalised, and

saving the new dimension values and the modified mappings in data storage means of the computer.

2. A method according to claim 1, wherein the step of creating new dimensional values and modifying the mapping comprises the steps of

examine whether the dimension is covering, i.e. that only immediate parent and child values can be related, as well as onto, i.e. that all paths in the hierarchy have equal lengths, and in case it is,

executing a make-strict procedure for making the dimension aggregation strict, i.e. that each child in a hierarchy has only one parent, thereby making the non-strict dimension aggregation normalised, i.e. summarisable, ~~the make-strict procedure being executed on the condition that the dimension is covering as well as onto prior to the execution~~.

3. A method according to claim 1 or 2, wherein the step of creating new dimensional values and modifying the mapping comprises the steps of

examine whether the dimension is covering, and in case it is,

executing a make-onto procedure for making the dimension onto, thereby at least partly making an ~~into~~non-onto dimension aggregation normalised~~, the make-onto procedure being executed on the condition that the dimension is covering prior to the execution~~.

4. A method according to any of claims 1-3, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the dimension covering, thereby at least partly making a non-covering dimension aggregation normalised.

5. A method according to any of claims 2-4, wherein the make-strict procedure comprises the steps of, starting from the bottom category and successively proceeding towards the top category,

identifying combinations of dimensional values of the same category for each of which combination at least one dimension value of a category below said category is linked to each of the dimension values of the combination,

creating one or more new dimensional values each representing one of the identified combinations of dimensional values and creating links from the new dimensional values to dimensional values of above categories in accordance with existing links from each of the dimensional values represented by the new dimensional value, and

identifying dimension values being linked to identified combinations of dimensional values of the same category and replacing the links with links to new dimensional values representing said combinations of dimensional values.

6. A method according to any of claims 2-5, wherein the make-strict procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category being a direct predecessor of the child category of which category at least one dimension value of the child category is linked to a dimension value of, setting said category as the parent category and performing the steps of:

(iia) ending the make-strict procedure for the parent category in case the parent category is the top category of the dimension,

(iib) ending the make-strict procedure for the parent category in case no dimension value of the parent category is linked to a dimension value of a higher category,

(iic) creating a new fused category in the dimension immediately below the parent category in case at least one of the dimension values of the child category is linked to more than one dimension value of the parent category,

(iid) for each dimensional value of the child category, performing the steps of: creating a new dimension value of the new fused category representing the one or more values of the parent category to which the dimensional value of the child category is linked and creating links from said new dimension value to said values in the parent category, the creation of the new dimension value being conditioned that no dimension value of the new fused category already exists having exactly such link(s), and

for each category being a direct predecessor of the parent category of which category at least one dimension value of the parent category is linked to a dimension value of, setting said category as a grandparent category and creating links from the new dimension value to the one or more dimension values of the grandparent category to which said one or more dimensional values of the parent category are linked,

(iie) removing the links from the parent category to the one or more grandparent categories, whereby the grandparent categories no longer are direct predecessors of the parent category,

(iif) creating links from each dimensional value of the child category to the dimension value of the new fused category having the same links to the dimension values of the parent category whereby the new fused category becomes a direct predecessor of the child category, and removing the links from the dimension values of the child category to the parent category, whereby the parent category no longer is a direct predecessor of the child category,

and

(iig) setting the new fused category as the child category and returning to step (ii).

7. A method according to any of claims 3-6, wherein the make-onto procedure comprises the steps of, starting from the top category and successively proceeding towards the bottom category,

creating, for each dimension value of each category above the bottom category not being linked to any dimensional value of the category immediately below, a new dimension value in the category immediately below and creating a link between said new dimension value and said dimension value of the category in question.

8. A method according to any of claims 3-7, wherein the make-onto procedure comprises the successive steps of

(i) setting the top category of the dimension as the parent category,

(ii) for each category immediately below the parent category and having dimension values being linked to dimension values of the parent category, setting said category as the child category and perform the steps of

(iia) creating, for each dimension value of the parent category not being linked to any dimensional value of the child category, a new dimension value in the child category and creating a link between said new dimension value and said dimension value of the parent category,

(iib) setting the child category as parent category,

(iic) ending the make-onto procedure in case the parent category is the bottom category of the dimension, else returning to step (ii) of the make-onto procedure.

9. A method according to any of claims 4-8, wherein the make-covering procedure comprises the successive steps of

identifying links between dimension values of two categories having at least one intermediate category there between,

creating a new dimension value in each of said intermediate categories for each of those links for which no paths of links exists going only through immediate child-parent links from lower to higher categories and including a link to a dimension value of the intermediate category, and

replacing those links with links between the dimension values of those links and the new dimension values.

10. A method according to any of claims 4-9, wherein the make-covering procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category immediately above the child category for which at least one link between a dimension value of said category and a dimension value of the child category exists, setting the category as the parent category and perform the steps of:

(iia) ending the make-covering procedure for the parent category in case the parent category is the top category of the dimension:,

(iib) for each higher category being a direct predecessor category of the child category and being higher in the hierarchy than the parent category, performing the steps of

(iiba) identifying sets of dimension values of the higher category and dimension values of the child category for which sets

a link exists, and

no paths of links going only from lower to higher categories and including a link to a dimension value of the parent category exists, and

(iibb) creating for each identified set of dimension values a new dimension value in the parent category, creating links between each of the dimension values of the set and the new dimension value, and removing the link between the two dimension values of the identified set, whereby the higher category no longer is a predecessor of the child category,

(iic) setting the parent category as the child category and returning to step (ii).

11. A method for by means of a computer to at least partly aggregation normalise a multidimensional object including a set of facts comprising a plurality of facts mapped on a plurality of dimensions having dimension values organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension, by means of applying the method of any of claims 1-10 to at least one of the dimensions of the multidimensional object.

12. A method according to claim 11, wherein the multidimensional object comprises a plurality of facts and the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, the facts constituting the bottom layer of each of the dimensions of the multidimensional object.

13. A method according to claim 11 or 12, comprising the steps of

selecting a subset of categories of the one or more dimension to be aggregation normalised, and

performing an aggregation normalisation of the selected subset,

whereby~~in~~ one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised~~., the performance of the partly aggregation normalisation being based on a selection of a subset of categories of the one or more dimension to be aggregation normalised.~~

14. A method according to any of claims 11-13, comprising the steps of

selecting specific aggregation functions to be performed on the multidimensional object, and

selecting by means of the computer normalisation steps to be performed based on the selection of specific aggregation functions to be performed,

whereby~~in~~ one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised. ~~, the normalisation steps to be performed being selected by means of the computer based on a selection of specific aggregation functions to be performed on the multidimensional object~~.

15. A method for by means of a computer to at least partly aggregation normalise a general on-line analytical processing multidimensional object including a set of facts comprising a plurality of facts mapped on an aggregation normalised plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension,

the method comprising the steps of

retrieve the mapping from data storage means associated with the computer,

including the mapping of the plurality of facts on the multidimensional object into the mapping of the multidimensional object so that the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, and the facts constitutes the bottom layer of each of the dimensions of the multidimensional object,

analysing the mapping of the multidimensional object to determine irregularities of the dimensions by means of analysing means executed by the computer,

creating new dimension values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is at least partly aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

16. A method according to claim 15, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-strict procedure for making the multidimensional object aggregation strict, thereby making the non-strict multidimensional object aggregation normalised, the make-strict procedure being executed on the condition that the multidimensional object is covering prior to the execution.

17. A method according to claim 15 or 16, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the multidimensional object covering, thereby at least partly making the non-covering multidimensional object aggregation normalised.

18. A method according to any of claims 15-17, wherein the method comprises the initial step of making each of the plurality of dimensions aggregation normalised by means of the method according to any of claims 1-10.

19. A method according to any of claims 11-18, wherein the created new dimensional values are marked as such, a pre-aggregation is performed on a multidimensional object being normalised by means of the computer ~~according to the method~~ and the method further comprises the step of

producing a reply to a query made to the system and concerning the multidimensional object, aggregate queries, exploring the dimension hierarchies, as well as navigation queries, that summarise the data at various levels of detail, in which reply the existence of the created new dimensional values is transparent.

20. A method according to any of claims 11-19, further comprising the steps of

implementing, into the aggregation normalised multidimensional object, of new facts including mapping of the facts onto the dimension, of new dimension values of the dimensions, or of new mapping between some of the dimension values, by which implementation irregularities of the multidimensional object is introduced,

analysing the introduced irregularities of the dimensions of the multidimensional object,

creating new dimensional values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

21. A computer system comprising at least one general purpose computer having data storage means associated therewith on which data storage means is stored a computer programme product suitable for adapting the computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20, the computer system comprising means for retrieving the computer programme product and perform accordingly.

22. A computer programme product suitable for adapting a general purpose computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

23. A computer system for on-line analytical processing having data storage means associated therewith on which a multidimensional object is stored, the multidimensional object including

a set of facts comprising a plurality of facts,

a first plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a first mapping of links between dimension values within each dimension of the first plurality of dimensions as well as links between the facts and the dimensions of the first plurality of dimensions, at least one of the dimensions of the first plurality of dimensions being irregular, and

a second plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a second mapping of links between dimension values within each dimension of the second plurality of dimensions as well as links between the facts and the dimensions of the second plurality of dimensions, each of the second plurality of dimensions being aggregation normalised,

the computer system comprising a query handler component being adapted for producing replies to queries made to the computer system and concerning the multidimensional object, the replies to navigation queries being based on the first set of dimensions and the replies to aggregate queries being based on the second set of dimensions.

24. A computer system according to claim 23, wherein a set of pre-aggregation data relating to the second plurality of dimensions is further stored within the data storage means and the replies to aggregate queries furthermore are based on the set of pre-aggregation data.

25. A computer system according to claim 23 or 24, wherein the query handler component is adapted for producing replies to aggregate queries in which replies the existence of the second plurality of dimensions is transparent.

26. A computer system according to claim 25, wherein the query handler component is adapted for transforming aggregate queries made to the first plurality of dimensions into queries for the second set of dimensions and transforming replies based on the second set of dimensions into replies as based on the first set of dimensions, thus making the existence of the second plurality of dimensions transparent in the produced reply.

27. A computer system according to claim 26, wherein the multidimensional object is stored within the data storage means of the computer system in tables organised as a combination of star schemes for the part of the multidimensional object containing only strict mappings, and additional tables containing the non-strict part of the mappings, the query handler component makes use of said tables in transforming queries and replies.

28. A computer system according to any of claims 23-27 further comprising means adapted for performing an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

Artist Bjørn Bjørnholt

The International Bureau of
World Intellectual Property Organization
34, chemin des Colombettes
CH-1211 Genève 20
Switzerland

BY TELEFAX (1/1 PAGE)
AND REGISTERED MAIL

- 8 JAN. 2002

Plougmann & Vingtoft

intellectual property consulting

Aarhus, 8 January 2002
International Patent Application No. PCT/DK00/00354
OLAP
Our ref: 25107 PC 01

Record of a Change

We hereby kindly request the record a change according to Rule 92*bis* PCT for the international patent application No. PCT/DK00/00354 entitled *Method and systems for making OLAP hierarchies summerisable*.

The change concerns the applicants:

Mindpass A/S is no longer an applicant for the above mentioned application and we kindly request the content of the register to be changed so that the following joint inventors and applicants are recorded as the sole joint applicants:

Torben Bach Pedersen,
Christian S. Jensen, and
Curtis E. Dyreson.

Please record this change and confirm safe receipt of this letter.

Yours sincerely,

Plougmann & Vingtoft a/s

JJS
Jens Jørgen Schmidt

Copenhagen

Plougmann & Vingtoft a/s
Sankt Annæ Plads 11
Post Office Box 3007
DK - 1021 Copenhagen K
CVR. No. 18 60 97 03
Telephone +45 33 63 93 00
Telefax +45 33 63 96 00
e-mail pv@pv.dk
www.pv.dk

Aarhus

Plougmann & Vingtoft
Arosgaarden
Åboulevarden 23
Post Office Box 49
DK - 8100 Aarhus C
Telephone +45 87 32 18 00
Telefax +45 33 63 96 00
e-mail pv@pv.dk
www.pv.dk

A subsidiary of Intellectual Property Holding a/s represented in
Copenhagen Aarhus Gothenburg
Helsingborg Linköping Lund
Malmö Ronneby Stockholm
Munich Reykjavik Alicante

combinations grows rapidly when the number of dimensions increase, while the sparseness of the multidimensional space decreases in higher dimension levels, meaning that aggregates at higher levels take up nearly as much space as lower-level aggregates. In some commercial applications, full pre-aggregation takes up as much as 200 times the space of the raw data [21]. Another problem with full pre-aggregation is that it takes too long to update the materialised aggregates when base data changes.

With the goal of avoiding data explosion, research has focused on how to select the best subset of aggregation levels given space constraints [1, 9, 11, 26, 28, 32] or maintenance time constraints [10], or the best combination of aggregate data and indices [8]. This approach is commonly referred to as *practical* (or partial or semi-eager [5, 11, 29]) pre-aggregation. Commercial OLAP systems now also exist that employ practical pre-aggregation, e.g., Microsoft Decision Support Services (Plato) [18] and Informix MetaCube [13].

The premise underlying the applicability of practical pre-aggregation is that lower-level aggregates can be *re-used* to compute higher-level aggregates, known as summarisability [16]. Summarisability occurs when the mappings in the dimension hierarchies are *onto* (all paths in the hierarchy have equal lengths), *covering* (only immediate parent and child values can be related), and *strict* (each child in a hierarchy has only one parent); and when also the relationships between facts and dimensions are many-to-one and facts are always mapped to the lowest levels in the dimensions [16]. However, the data encountered in many real-world applications fail to comply with this rigid regime. This motivates the search for techniques that allow practical pre-aggregation to be used for a wider range of applications, the focus of the present invention.

Description of the Invention

Motivated by the increasing use of OLAP systems in many different applications, including in business and health care, the present invention provides transformation techniques for multidimensional databases that leverage the existing, performance-enhancing technique, known as practical, or partial or semi-eager, pre-aggregation, by making this technique relevant to a much wider range of real-world applications.

Current pre-aggregation techniques assume that the dimensional structures are *summarisable*. Specifically, the mappings in dimension hierarchies must be *onto*,

Detailed description of the Invention

We now proceed to describe the invention in detail. The next section presents a real-world clinical case study that exemplifies the non-summarisable properties of real-world applications. The following section proceeds to define the aspects of a multidimensional data model necessary for describing the new techniques, and defines also important properties related to summarisability. Algorithms are presented for transforming dimension hierarchies to achieve summarisability, then apply the algorithms to fix non-summarisable relationships between facts and dimensions. It is also demonstrated how the techniques may be integrated into current systems, transparently to the user and how to modify the algorithms to accommodate incremental computation.

properties, so applying each once is sufficient.

In general, the algorithms take as input a set of tables R_{C_1,C_2} that specifies the mapping from dimension values in category C_1 to values in category C_2. The input needs not contain all pairs of ancestors and descendants—only direct parent-child relationships are required. If there are non-covering mappings in the hierarchy, we have categories C, P, H such that $\{P, H\} \subseteq \mathit{Pred}(C)$ and $\mathit{Type}(P) < \mathit{Type}(H)$. In this case, the input must also contain $R_{P,H}$ tables that map P values to H values.

The algorithms are expressed using recursion. They could also easily be expressed using iteration instead.

Non-Covering Hierarchies

The first algorithm renders all mappings in a dimension hierarchy covering w.r.t. any category. When a dimension value is mapped *directly* to another value in a category higher than the one immediately above it in the hierarchy, a new intermediate value is inserted into the category immediately above, and the two original dimension values are linked to this new value, rather than to each other.

Example 6 In the hierarchy for the Residence dimension, two links go from Address directly to County. The address "123 Rural Road" (52) is in "Melbourne County" (31), but not in a city, and the address "1 Sandy Dunes" (53) is in "Outback County" (32), which does *not* have any cities at all. The algorithm inserts two new dimension values in the City category, **C31** and **C32**, which represent Melbourne and Outback county, respectively, and links them to their respective counties. The addresses "123 Rural Road" and "1 Sandy Dunes" are then linked to **C31** and **C32**, respectively. This occurs in the first call of procedure MakeCovering (on the Address category; the procedure is given below). When MakeCovering is called recursively on the City, County, and $\top$ categories, nothing happens, as all mappings are already covering. The transformation is illustrated graphically in Figure 3. The dotted lines show the "problematic" links, and the bold-face values and thick lines show the new dimension values and links.

In the algorithm, C is a *child* category, P is an immediate *parent* category, H is a "*higher*" category, L are the non-covering *links* from C to H, and N are the "higher" dimension values in L. The $\bowtie$ operator denotes natural join. The algorithm works as follows. Given the argument category C (initially the bottom category) in line (1), the algorithms goes through all C's

Non-Onto Hierarchies

The second algorithm renders all mappings in hierarchies onto, i.e., all dimension values in non-bottom categories have children. This is ensured by inserting placeholder values in lower categories to represent the childless values. These new values are marked with the original values, making it possible to map facts to the new placeholder values instead of to the original values. This makes it possible to only map facts to the bottom category.

Example 7 In the Diagnosis dimension, the "Lung cancer" diagnosis family (ID = 14) has no children. When the algorithm reaches the Diagnosis Family category, it inserts a placeholder value (**L14**) into the Low-level Diagnosis category, representing the "Lung cancer" diagnosis, and links it to the original value. Facts mapped to the "Lung cancer" value may then instead be mapped to the new placeholder value, ensuring that facts are mapped only to the Low-level Diagnosis Category. A graphical illustration of the transformation is seen in Figure 4. The bold-faced **L14** value is the new value inserted, and the thick line between 14 and **L14** is the new link inserted.

In the algorithm below, P is a *parent* category, C is a *child* category, and N holds the parent values with *no* children. The algorithm works as follows. Given a category P (initially the $\top$ category) in line (1), the algorithm goes through all categories C that are (immediate) descendants of P (2). For each C, line (4) finds the values N in P that have *no* children in C, by "subtracting" the values *with* children in C from the values in P. For each "childless" value in N, lines (5) and (6), respectively, insert into C a placeholder value marked with the parent value, and links the new value to the original. MakeOnto is then called recursively on C (7). The algorithms terminates when it reaches the $\bot$ category, which has no descendants.

```
(1)  procedure MakeOnto(P)
(2)      for each C ∈ Desc(P) do
(3)      begin
(4)          N ← P \ Π_P(R_{C,P})
(5)          C ← C ∪ {Mark(p) | p ∈ N}
(6)          R_{C,P} ← R_{C,P} ∪ {(Mark(p), p) | p ∈ N}
(7)          MakeOnto(C)
(8)      end
(9)  end
```

ure 5. Because of the non-strictness in the mapping from Low-level Diagnosis to Diagnosis Family, and from Diagnosis Family to Diagnosis Group, two new category types and the corresponding categories are introduced. The third picture indicates the argument to the algorithm; and, in addition, its dotted lines indicate the links deleted by the algorithm. The fourth picture gives the result of applying the algorithm; here, the bold-face values and thick lines indicate the values and links inserted by the algorithm.

In the first call of the algorithm the three Low-level Diagnosis values—"(low-level) Lung cancer" (L14); "Insulin dependent diabetes during pregnancy" (5); and "Non insulin dependent diabetes during pregnancy" (6)—are linked to the three new fused values—"(low-level) Lung cancer" (**14**); "Diabetes during pregnancy, Insulin dependent diabetes" (**4, 9**); and "Diabetes during pregnancy, Non insulin dependent diabetes" (**4, 10**)—and these are in turn linked to "Lung Cancer" (14); "Diabetes during pregnancy" (4); "Insulin dependent diabetes" (9); and "Non insulin dependent diabetes" (10). The these latter four values in the Diagnosis Family category are un-linked from their parents, as the Diagnosis Family category is "unsafe."

When called recursively on the Set-of Diagnosis Family category, the algorithm creates the new fused values "Cancer" (**13**) and "Diabetes, Other pregnancy related diseases" (**11, 12**) in the Set-of Diagnosis Group category. These new values are linked to the values "Cancer" (13), "Diabetes" (11), and "Other pregnancy related diseases" (12) in the Diagnosis Group category, and to the $\top$ value; and the values in the Diagnosis Group category are un-linked from their parents. Note the importance of having a $\top$ value: the values not linked to $\top$ are exactly the unsafe values, for which aggregate results should not be re-used.

The algorithm assumes that all paths in the dimension hierarchy have equal length, i.e., all direct links are from children to their immediate parents. This is ensured by the MakeCovering and MakeOnto algorithms. In the algorithm below, C is a *child* category, P is a *parent* category, G is a *grandparent* category, N is the *new* category introduced to hold the "fused" values, and $\bowtie$ denotes natural join.

The algorithm takes a category C (initially the $\bot$ category) as input. I then goes through the set of immediate parent categories P of C (line (2)). Line (4) tests if there is non-strictness in the mapping from C to P *and* if P has any parents (4). If this test fails, there is no problem as aggregate results for P can either be safely re-used or are guaranteed not be re-used; and the algorithm in then invoked recursively, in line (20).

If the test succeeds, the algorithm creates a new fused category. First, a new, empty category N with domain 2^P is created in line (6). The values inserted into this category represent *sets* of values of P. For example. the value "**1, 2**" represents the set consisting of

F, which is now the bottom of the lattice. The algorithm makes the mappings covering w.r.t. the facts by inserting new marked values, representing the parent values, in the intermediate categories, and by linking the facts to the new values instead of the parent values. As in the section "Non-Covering Hierarchies," the marked values keep information about their original values, so that when new fact-dimension mappings are added, the links that are supposed to go *directly* to the original parent values now instead can be set to go to the marked value in the $\perp$ category.

Example 9 In the case study, the mapping between Patients and Diagnoses is of mixed granularity: "John Doe" (1) and "Jane Doe" are both mapped to the Diagnosis Family, "Insulin dependent diabetes" (9), "Jane Doe" is additionally mapped to the Low-level Diagnosis, "Insulin dependent diabetes during pregnancy" (5), and "Jim Doe" is mapped to "Diabetes" (11), a Diagnosis Group.

In the first call of the algorithm, two new Low-level Diagnoses are inserted: "**L9**," representing "Insulin dependent diabetes," and "**L11**," representing "Diabetes"; and the facts are mapped to these instead of the original values. In the recursive call on Low-level Diagnosis, an "**F11**" value representing "Diabetes" at the Diagnosis Family level is inserted between "Diabetes" and value "**L11**."

The transformations are illustrated in Figures 6 and 7, where dotted lines indicate links that are deleted by the algorithm and bold-face value and thick lines indicate dimension values and links inserted by the algorithm.

Many-To-Many Relationships

The second case occurs when relationships between facts and dimension values are many-to-many. This means that the hierarchy, with the facts as the bottom category, is non-strict, leading to possible double-counting of facts. It is enough to make the hierarchy partly strict, as described in the section "Non-Strict Hierarchies." The MakeStrict algorithm is initially called on F, which is now the bottom of the hierarchy lattice. Because the MakeCovering algorithm has already been applied, all paths from facts to the $\top$ value have equal length, as required by the MakeStrict algorithm.

Some dimension values have no facts mapped to them, leading to an interesting side effect of the algorithm. When the algorithm fuses values and places the fused values in-between the original values, it also deletes the child-to-parent and parent-to-grandparent links. The

fact-less dimension values are then left disconnected from the rest of the hierarchy, with no links to other values.

These fact-less dimension values do not contribute to any aggregate computations and are thus superfluous. To minimise the dimensions, an "Delete-unconnected" algorithm that deletes the fact-less dimension values by traversing the hierarchy starting at the facts is invoked in a postprocessing step. For a hierarchy of height k, this can be done in time $\mathcal{O}(kn \log n)$, where n is the size of the mapping between facts and dimensions. Thus, the overall computational complexity is not altered.

Example 10 The relationship between patients and diagnoses is many-to-many. In Example 9, the MO was transformed so that all mappings were covering, as seen in Figure 6; algorithm MakeStrict is applied to this MO. The final result of the application of the algorithm is seen in Figure 7. Values in italics, e.g., *L14*, and dotted lines indicate deleted values and links. Bold-face values and thick lines denote values and links inserted by the algorithm.

Three new categories are introduced: "Set-of Low-level Diagnosis," "Set-of Diagnosis Family," and "Set-of Diagnosis Group," as non-strictness occurs at all levels. Fused values are inserted into these fused categories. For example, values "(low-level) Lung Cancer" (**L14**), "Insulin dependent diabetes during pregnancy, (low-level) Insulin dependent diabetes" (**5**, **L9**), and "(low-level) Insulin dependent diabetes" (**L9**) are inserted into the "Set-of Low-level Diagnosis" category; and the original values are linked to the new values.

Values "(low-level) Lung cancer" (**L14**), "Lung cancer" (14), "Cancer" (13), "Non insulin dependent diabetes during pregnancy" (6), and "Non insulin dependent diabetes" (10) do not characterise any facts and are deleted by "Delete-unconnected."

Architectural Context

Overview

The overall idea presented in this paper is to take un-normalised MOs and transform them into normalised MOs that are well supported by the practical pre-aggregation techniques available in current OLAP systems. Queries are then evaluated on the transformed MOs. However, we still want the users to see only the original MOs, as they reflect the users' understanding of the domain. This prompts the need for means of handling both the original and the transformed

MOs. This section explores this coexistence.

A current trend in commercial OLAP technology is the separation of the front-end presentation layer from the back-end database server. Modern OLAP applications consist of an OLAP client that handles the user interface and an OLAP server that manages the data and processes queries. The client communicates with the server using a standardised application programming interface (API), e.g., Microsoft's OLE DB for OLAP [17] or the OLAP Council's MDAPI [20]. The architecture of such a system is given to the left in Figure 8.

This separation of client and server facilitates our desire to have the user see the original MO while queries are evaluated against the transformed MO. Studies have shown that queries on a data warehouse consist of 80% *navigational* queries that explore the dimension hierarchies and 20% *aggregation* queries that summarise the data at various levels of detail [14]. Examples of navigational and aggregation queries are "Show me the Low-Level Diagnoses contained in the Insulin-Dependent Diabetes Diagnosis Family" and "Show me the count of patients, grouped by Diagnosis Family," respectively. The navigational queries must be performed on the *original* MO, while the aggregation queries must be performed on the *transformed* MO. This is achieved by introducing an extra "Query Handler" component between the client and the server. The OLAP client sends a query to the query handler, the primary task of which is to determine whether the query is a navigational query (internal to a dimension) or an aggregation query (involving the facts). Navigational queries are passed to one OLAP server that handles the original (navigational) data, while aggregation queries are passed to another OLAP server that manages the transformed (aggregation) data. This extended system architecture is seen to the right in Figure 8.

The OLAP server for navigation data needs to support dimension hierarchies which have non-summarisable properties, a requirement not yet supported by many commercial systems today. However, relational OLAP systems using snow-flake schemas [14] are able to support this type of hierarchies, as are some other OLAP systems, e.g., Hyperion (Arbor) Essbase [12]. If the OLAP system available does not have sufficiently flexible hierarchy support, one solution is to build a special-purpose OLAP server that conforms to the given API. This task is not as daunting as it may seem at first because only *navigational* queries need to be supported, meaning that multidimensional queries can be translated into simple SQL "lookup" queries.

We note that the only data needed to answer navigational queries is the hierarchy definitions. Thus, we only need to store the fact data (facts and fact-dimension relations, in our model) once, in the aggregational data, meaning that the overall storage requirement is only slightly larger than storing just the aggregational data. Navigational queries are evaluated on

the original hierarchy definitions and do not need to be re-written by the query handler.

As described in the section "Dimension Transformation Techniques," aggregation queries need to be re-written slightly by adding an extra HAVING clause condition to exclude results for the new values inserted by the transformation algorithms. This can easily be done automatically by the query handler, giving total transparency for the user. Even though the added HAVING clause conditions are only necessary for the covering and onto transformations, they can also be applied to hierarchies transformed to achieve strictness; this has no effect, but simplifies the query rewriting. The new values can also be filtered out using a modified WHERE clause, by performing an inner join with a table containing only the original values, or by using nested SELECT statements as described in the next section.

Concrete Implementation

We now show how the abstract architecture described above can be implemented using standard relational database technology.

The transparency is achieved by working with two versions of each user-specified hierarchy and by using a query rewrite mechanism. This is described in detail later in this section. The overall system architecture is seen in Figure 9.

DiagID	Lowlevel	Family	Group
5	Insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Diabetes
5	Insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Pregnancy related
5	Insulin dependent diabetes during pregnancy	Insulin dependent diabetes	Diabetes
6	Non insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Diabetes
6	Non insulin dependent diabetes during pregnancy	Diabetes during pregnancy	Pregnancy related
6	Non insulin dependent diabetes during pregnancy	Non insulin dependent diabetes	Diabetes
100	!Lowlevel!Lung Cancer	Lung cancer	Cancer

Table 2: DDiagnosis Dimension Table

The ROLAP client tool, in this case the ROLAP tool Synchrony, which originated from Kimball's Startracker tool [14], makes SQL requests to the ROLAP database, in this case the Oracle8 RDBMS, using the ODBC standard. We have implemented a special, query-transforming ODBC driver (QTOD) that, based on case-specific metadata, transforms the SQL requests into requests that hide the transformations from the users, returning the query results that the user would expect based on the original hierarchies. A transformed request is submitted to the OLAP DB using an RDBMS-specific ODBC driver. The QTOD component is common to all RDBMSs, so Oracle8 may be replaced by another RDBMS such as IBM DB2, Informix, or MS SQL Server. Another ROLAP tool may also be used, making the solution

```
SELECT DISTINCT Lowlevel
FROM DDiagnosis
WHERE Lowlevel NOT LIKE '!%'
```

Lowlevel
Insulin dependent diabetes during pregnancy
Non insulin dependent diabetes during pregnancy

Table 3: Navigational Query Result

Due to the use of `DISTINCT` as a quantifier, duplicates are not returned. The `NOT LIKE` predicate removes the placeholder values inserted into the hierarchy to balance it, which in this case is the value "!Lowlevel!Lung Cancer." As desired, the result is unaffected by the translations.

For *aggregation queries*, it is also possible to achieve transformation transparency, although this is more difficult. For dimensions with non-strictness, a special dimension table is introduced that holds only the part of the normalised hierarchy that does *not* contain non-strictness. In the normalised hierarchy to the right in Figure 5, this part is the Low-level Diagnosis category and the two special categories introduced by the normalisation process to hold *sets of diagnosis families* and *sets of diagnosis groups*, respectively. This part of the hierarchy is implemented in the Diagnosis dimension table seen in Table 4.

DiagID	Lowlevel	Family	Group
1000020	!Low-level Diagnosis!Lung cancer	14	13
5	Insulin dependent diabetes during pregnancy	4,9	11,12
6	Non insulin dependent diabetes during pregnancy	4,10	11,12

Diagnosis

Group	SGroup
Cancer	13
Diabetes	11,12
Pregnancy Related	11,12

SGroup

Table 4: Dimension and Group Tables for Aggregation

The "Lowlevel" column contains the normal textual diagnosis description, whereas the

special "Family" and "Group" columns contain comma-separated ordered lists of the IDs of the sets of values that are represented by the column values. For example, value "4,9" represents the set $\{4, 9\}$.

We need to capture the remaining part of the hierarchy, which consists of non-strict mappings from a "set-of-X" category to the "X" category, e.g., the mapping of the "set-of-Diagnosis Group" category to the "Diagnosis Group" category to the right in Figure 5, which maps $\{13\}$ to 13 (Cancer) and $\{11, 12\}$ to 11 (Diabetes) and 12 (Pregnancy Related). This is done by introducing a special table for each such mapping, named by the category prefixed with an "S" (for Set-of). For example, for the Diagnosis Group category, table "SGroup" in Table 4 maps sets of diagnosis groups to the individual diagnosis groups in the sets. The "Group" column represents the diagnosis group, while the "SGroup" column represents the associated set of diagnosis groups.

With these tables available, it is possible to obtain transformation transparency for aggregation queries. A ROLAP aggregation query has the format of the query below that computes the number of patients per diagnosis group.

```
SELECT Diagnosis.Group, SUM(Patient.Count)
FROM Diagnosis,Patient
WHERE Diagnosis.DiagID=Patient.DiagID
GROUP BY Diagnosis.Group
```

This is transformed into the more complex query given next.

```
SELECT SGroup.Group, SUM(QQQQQQQ.Count)
FROM Sgroup,
 (SELECT Diagnosis.Group,
         SUM(Patient.Count) AS Count
  FROM Diagnosis,Patient
  WHERE Diagnosis.DiagID=Patient.DiagID
  GROUP BY Diagnosis.Group) QQQQQQQ
WHERE QQQQQQQ.Group=SGroup.SGroup AND
      SGroup.SGroup NOT LIKE '!%'
GROUP BY SGroup.Sgroup
```

The transformed aggregation query has two parts. The nested table expression computes the number of patients per *set of diagnosis group*, making this available via correlation name QQQQQQQ. This part of the hierarchy is a balanced tree, so the RDBMS can safely use pre-aggregated data for optimising the query performance. The result of the nested table

CLAIMS

1. A method for by means of a computer to at least partly aggregation normalise a dimension having dimension values organised into categories of dimension values based on a partial ordering, the dimension comprising mappings of links between dimension values, the method comprising the steps of

retrieve the mapping from data storage means associated with the computer,

analysing the mapping to determine irregularities of the dimension by means of analysing means executed by the computer,

creating new dimension values of the dimension and modifying the mapping between dimensional values of the dimension according to the analysis, whereby the dimension is at least partly aggregation normalised, and

saving the new dimension values and the modified mappings in data storage means of the computer.

2. A method according to claim 1, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-strict procedure for making the dimension aggregation strict, thereby making the non-strict dimension aggregation normalised, the make-strict procedure being executed on the condition that the dimension is covering as well as onto prior to the execution.

3. A method according to claim 1 or 2, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-onto procedure for making the dimension onto, thereby at least partly making an into dimension aggregation normalised, the make-onto procedure being executed on the condition that the dimension is covering prior to the execution.

4. A method according to any of claims 1-3, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the dimension covering, thereby at least partly making a non-covering dimension aggregation normalised.

5. A method according to any of claims 2-4, wherein the make-strict procedure comprises the steps of, starting from the bottom category and successively proceeding towards the top category,

identifying combinations of dimensional values of the same category for each of which combination at least one dimension value of a category below said category is linked to each of the dimension values of the combination,

creating one or more new dimensional values each representing one of the identified combinations of dimensional values and creating links from the new dimensional values to dimensional values of above categories in accordance with existing links from each of the dimensional values represented by the new dimensional value, and

identifying dimension values being linked to identified combinations of dimensional values of the same category and replacing the links with links to new dimensional values representing said combinations of dimensional values.

6. A method according to any of claims 2-5, wherein the make-strict procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category being a direct predecessor of the child category of which category at least one dimension value of the child category is linked to a dimension value of, setting said category as the parent category and performing the steps of:

(iia) ending the make-strict procedure for the parent category in case the parent category is the top category of the dimension,

(iib) ending the make-strict procedure for the parent category in case no dimension value of the parent category is linked to a dimension value of a higher category,

(iic) creating a new fused category in the dimension immediately below the parent category in case at least one of the dimension values of the child category is linked to more than one dimension value of the parent category,

(iid) for each dimensional value of the child category, performing the steps of: creating a new dimension value of the new fused category representing the one or more values of the parent category to which the dimensional value of the child category is linked and creating links from said new dimension value to said values in the parent category, the creation of the new dimension value being conditioned that no dimension value of the new fused category already exists having exactly such link(s), and

for each category being a direct predecessor of the parent category of which category at least one dimension value of the parent category is linked to a dimension value of, setting said category as a grandparent category and creating links from the new dimension value to the one or more dimension values of the grandparent category to which said one or more dimensional values of the parent category are linked,

(iie) removing the links from the parent category to the one or more grandparent categories, whereby the grandparent categories no longer are direct predecessors of the parent category,

(iif) creating links from each dimensional value of the child category to the dimension value of the new fused category having the same links to the dimension values of the parent category whereby the new fused category becomes a direct predecessor of the child category, and removing the links from the dimension values of the child category to the parent category, whereby the parent category no longer is a direct predecessor of the child category,

and

(iig) setting the new fused category as the child category and returning to step (ii).

7. A method according to any of claims 3-6, wherein the make-onto procedure comprises the steps of, starting from the to category and successively proceeding towards the bottom category,

creating, for each dimension value of each category above the bottom category not being linked to any dimensional value of the category immediately below, a new dimension value in the category immediately below and creating a link between said new dimension value and said dimension value of the category in question.

8. A method according to any of claims 3-7, wherein the make-onto procedure comprises the successive steps of

(i) setting the top category of the dimension as the parent category,

(ii) for each category immediately below the parent category and having dimension values being linked to dimension values of the parent category, setting said category as the child category and perform the steps of

(iia) creating, for each dimension value of the parent category not being linked to any dimensional value of the child category, a new dimension value in the child

category and creating a link between said new dimension value and said dimension value of the parent category,

(iib) setting the child category as parent category,

(iic) ending the make-onto procedure in case the parent category is the bottom category of the dimension, else returning to step (ii) of the make-onto procedure.

9. A method according to any of claims 4-8, wherein the make-covering procedure comprises the successive steps of

identifying links between dimension values of two categories having at least one intermediate category there between,

creating a new dimension value in each of said intermediate categories for each of those links for which no paths of links exists going only through immediate child-parent links from lower to higher categories and including a link to a dimension value of the intermediate category, and

replacing those links with links between the dimension values of those links and the new dimension values.

10. A method according to any of claims 4-9, wherein the make-covering procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category immediately above the child category for which at least one link between a dimension value of said category and a dimension value of the child category exists, setting the category as the parent category and perform the steps of:

(iia) ending the make-covering procedure for the parent category in case the parent category is the top category of the dimension:,

(iib) for each higher category being a direct predecessor category of the child category and being higher in the hierarchy than the parent category, performing the steps of

(iiba) identifying sets of dimension values of the higher category and dimension values of the child category for which sets

a link exists, and

no paths of links going only from lower to higher categories and including a link to a dimension value of the parent category exists, and

(iibb) creating for each identified set of dimension values a new dimension value in the parent category, creating links between each of the dimension values

of the set and the new dimension value, and removing the link between the two dimension values of the identified set, whereby the higher category no longer is a predecessor of the child category,

(iic) setting the parent category as the child category and returning to step (ii).

11. A method for by means of a computer to at least partly aggregation normalise a multidimensional object including a set of facts comprising a plurality of facts mapped on a plurality of dimensions having dimension values organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension, by means of applying the method of any of claims 1-10 to at least one of the dimensions of the multidimensional object.

12. A method according to claim 11, wherein the multidimensional object comprises a plurality of facts and the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, the facts constituting the bottom layer of each of the dimensions of the multidimensional object.

13. A method according to claim 11 or 12, wherein one or more of the dimensions of the multidimensional object is only partly aggregation normalised, the performance of the partly aggregation normalisation being based on a selection of a subset of categories of the one or more dimension to be aggregation normalised.

14. A method according to any of claims 11-13, wherein one or more of the dimensions of the multidimensional object is only partly aggregation normalised, the normalisation steps to be performed being selected by means of the computer based on a selection of specific aggregation functions to be performed on the multidimensional object.

15. A method for by means of a computer to at least partly aggregation normalise a multidimensional object including a set of facts comprising a plurality of facts mapped on an aggregation normalised plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension,

the method comprising the steps of

retrieve the mapping from data storage means associated with the computer,

including the mapping of the plurality of facts on the multidimensional object into the mapping of the multidimensional object so that the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, and the facts constitutes the bottom layer of each of the dimensions of the multidimensional object,

analysing the mapping of the multidimensional object to determine irregularities of the dimensions by means of analysing means executed by the computer,

creating new dimension values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is at least partly aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

16. A method according to claim 15, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-strict procedure for making the multidimensional object aggregation strict, thereby making the non-strict multidimensional object aggregation normalised, the make-strict procedure being executed on the condition that the multidimensional object is covering prior to the execution.

17. A method according to claim 15 or 16, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the multidimensional object covering, thereby at least partly making the non-covering multidimensional object aggregation normalised.

18. A method according to any of claims 15-17, wherein the method comprises the initial step of making each of the plurality of dimensions aggregation normalised by means of the method according to any of claims 1-10.

19. A method according to any of claims 11-18, wherein the created new dimensional values are marked as such, a pre-aggregation is performed on a multidimensional object being normalised by means of the computer according to the method and the method further comprises the step of

producing a reply to a query made to the system and concerning the multidimensional object, aggregate queries as well as navigation queries, in which reply the existence of the created new dimensional values is transparent.

20. A method according to any of claims 11-19, further comprising the steps of

implementing, into the aggregation normalised multidimensional object, of new facts including mapping of the facts onto the dimension, of new dimension values of the dimensions, or of new mapping between some of the dimension values, by which implementation irregularities of the multidimensional object is introduced,

analysing the introduced irregularities of the dimensions of the multidimensional object,

creating new dimensional values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

21. A computer system comprising at least one general purpose computer having data storage means associated therewith on which data storage means is stored a computer programme product suitable for adapting the computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20, the computer system comprising means for retrieving the computer programme product and perform accordingly.

22. A computer programme product suitable for adapting a general purpose computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

23. A computer system having data storage means associated therewith on which a multidimensional object is stored, the multidimensional object including

a set of facts comprising a plurality of facts,

a first plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a first mapping of links between dimension values within each dimension of the first plurality of dimensions as well as links between the facts and the dimensions of the first plurality of

dimensions, at least one of the dimensions of the first plurality of dimensions being irregular, and

a second plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a second mapping of links between dimension values within each dimension of the second plurality of dimensions as well as links between the facts and the dimensions of the second plurality of dimensions, each of the second plurality of dimensions being aggregation normalised,

the computer system comprising a query handler component being adapted for producing replies to queries made to the computer system and concerning the multidimensional object, the replies to navigation queries being based on the first set of dimensions and the replies to aggregate queries being based on the second set of dimensions.

24. A computer system according to claim 23, wherein a set of pre-aggregation data relating to the second plurality of dimensions is further stored within the data storage means and the replies to aggregate queries furthermore are based on the set of pre-aggregation data.

25. A computer system according to claim 23 or 24, wherein the query handler component is adapted for producing replies to aggregate queries in which replies the existence of the second plurality of dimensions is transparent.

26. A computer system according to claim 25, wherein the query handler component is adapted for transforming aggregate queries made to the first plurality of dimensions into queries for the second set of dimensions and transforming replies based on the second set of dimensions into replies as based on the first set of dimensions, thus making the existence of the second plurality of dimensions transparent in the produced reply.

27. A computer system according to claim 26, wherein the multidimensional object is stored within the data storage means of the computer system in tables organised as a combination of star schemes for the part of the multidimensional object containing only strict mappings, and additional tables containing the non-strict part of the mappings, the query handler component makes use of said tables in transforming queries and replies.

28. A computer system according to any of claims 23-27 further comprising means adapted for performing an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70) 14

Applicant's or agent's file reference	FOR FURTHER ACTION	See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)
25107 PC 1		
International application No.	International filing date *(day/month/year)*	Priority date *(day/month/year)*
PCT/DK00/00354	30/06/2000	21/07/1999
International Patent Classification (IPC) or national classification and IPC G06F17/30		
Applicant MINDPASS A/S et al.		

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 5 sheets, including this cover sheet.

   ☒ This report is also accompanied by ANNEXES, i.e. sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

   These annexes consist of a total of 22 sheets.

3. This report contains indications relating to the following items:

   - I ☒ Basis of the report
   - II ☐ Priority
   - III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
   - IV ☐ Lack of unity of invention
   - V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations suporting such statement
   - VI ☐ Certain documents cited
   - VII ☒ Certain defects in the international application
   - VIII ☐ Certain observations on the international application

Date of submission of the demand	Date of completion of this report
29/01/2001	03.08.2001
Name and mailing address of the international preliminary examining authority: European Patent Office D-80298 Munich Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Authorized officer Nippl, C Telephone No. +49 89 2399 7372

Form PCT/IPEA/409 (cover sheet) (January 1994)

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No. PCT/DK00/00354

I. Basis of the report

1. With regard to the **elements** of the international application *(Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to this report since they do not contain amendments (Rules 70.16 and 70.17))*:
Description, pages:

1-48 as originally filed

Claims, No.:

1-28 as originally filed

Drawings, sheets:

1/4-4/4 as originally filed

2. With regard to the **language**, all the elements marked above were available or furnished to this Authority in the language in which the international application was filed, unless otherwise indicated under this item.

These elements were available or furnished to this Authority in the following language: , which is:

- ☐ the language of a translation furnished for the purposes of the international search (under Rule 23.1(b)).
- ☐ the language of publication of the international application (under Rule 48.3(b)).
- ☐ the language of a translation furnished for the purposes of international preliminary examination (under Rule 55.2 and/or 55.3).

3. With regard to any **nucleotide and/or amino acid sequence** disclosed in the international application, the international preliminary examination was carried out on the basis of the sequence listing:

- ☐ contained in the international application in written form.
- ☐ filed together with the international application in computer readable form.
- ☐ furnished subsequently to this Authority in written form.
- ☐ furnished subsequently to this Authority in computer readable form.
- ☐ The statement that the subsequently furnished written sequence listing does not go beyond the disclosure in the international application as filed has been furnished.
- ☐ The statement that the information recorded in computer readable form is identical to the written sequence listing has been furnished.

4. The amendments have resulted in the cancellation of:

- ☐ the description, pages:
- ☐ the claims, Nos.:

☐ the drawings, sheets:

5. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed (Rule 70.2(c)):

(Any replacement sheet containing such amendments must be referred to under item 1 and annexed to this report.)

6. Additional observations, if necessary:

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Yes:	Claims	1-28
	No:	Claims	
Inventive step (IS)	Yes:	Claims	1-28
	No:	Claims	
Industrial applicability (IA)	Yes:	Claims	1-28
	No:	Claims	

2. Citations and explanations
see separate sheet

VII. Certain defects in the international application

The following defects in the form or contents of the international application have been noted:
see separate sheet

Re Item V

Reasoned statement under Rule 66.2(a)(ii) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Reference is made to the following documents:

   D1: JIM GRAY ET AL: 'Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals' DATA MINING AND KNOWLEDGE DISCOVERY 1, 1997, pages 29-53, XP002901286

   D2: H.J. LENZ ET AL: 'Summarizability in OLAP and Statistical Data bases' SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 1997. PROCEEDINGS, NINTH INTERNATIONAL CONFERENCE , pages 132-143, XP002901287

   D3: INDERPAL SINGH MUMICK: 'Maintenance of Data Cubes and Summary Tables in a Warehouse' AT&T LABORATORIES, [Online] 1997, XP002901288

2. Independent claims 1, 15 and 23 relate to the problem of modifying existing dimension hierarchies in a multidimensional data space in order to provide summarisability.
   Summarisability is an important property in statistical and OLAP applications, because violating this condition can lead to erroneous conclusions and decisions when using aggregates.

2.1 Document D1, which is considered to represent the most relevant state of the art, discloses a new database operator that generalizes aggregations for the N- dimensional data space. Solutions are proposed on how to integrate this operator on the execution and (SQL) language level.
   D2 discloses a method for testing the summarisability condition, without proposing a solution on how to establish this condition.
   D3 deals with maintenance problems of aggregates in the case of updates.

2.3 In document D1 it is mentioned that irregular dimension hierarchies render pre-aggregation impossible but no solution to this problem is provided.

In claims 1,15 and 23 of the present application this problem is solved by creating new dimension values and modifying the mapping among dimension values accordingly. This method enables pre-aggregates also for dimension hierarchies which are not covering, onto and strict.

Additionally, claim 23 sets out a method on how to keep this procedure transparent to the user by using two sets of dimensions.

This solution is not known nor rendered obvious by the available prior art and is thus considered as involving an inventive step (Article 33(3) PCT).

Re Item VII

Certain defects in the international application

1. The description (summary of the invention) is not in conformity with the claims as required by Rule 5.1(a)(iii) PCT.

PLOUGMANN VINGTOFT & PARTNERS

International Preliminary Examining Authority
European Patent Office
Erhardtstrasse 27
D-80298 München
Germany

12 JUNI 2001

BY TELEFAX (1/25 pages)
AND CONFIRMATION BY MAIL

PCT CHAPTER II

12 June 2001

International Patent Application No. PCT/DK00/00354
OLAP
Our ref: 25107 PC 01

COPENHAGEN OFFICE:
SANKT ANNÆ PLADS 11
POST OFFICE BOX 3007
DK - 1021 COPENHAGEN K
A/S REG. NO. 223795
TELEPHONE +45 33 63 93 00
TELEFAX +45 33 63 96 00
e-mail pv@pv.dk
www.pv.dk

Artist Bjørn Bjørnholt

AARHUS OFFICE:
NOBELPARKEN
JENS CHR. SKOUS VEJ 9
POST OFFICE BOX 49
DK - 8100 AARHUS C
TELEPHONE +45 87 39 18 00
TELEFAX +45 33 63 96 00
e-mail pv@pv.dk
www.pv.dk

Referring to the Written Opinion dated 14 March 2110, we hereby submit a set of amended claims, a set of amended claims with the amendments indicated and new description pages, all amendments being carried out in accordance with the recommendations of the Examiner.

Clarity of the claims, Art. 6 PCT

The Examiner states in Item VIII that the claims 1-28 are not clear with specific reference to PCT Guidelines III-4.2, because the meaning of certain words is not well-defined. We have by amendments clarified the technical field of the scope of protection and thereby the technical terms and we have added descriptive matter from the description to the wording of the claims. We therefore respectfully submit that the amended claim are clear and in accordance with Art. 6 PCT and that the amendments are all firmly based on the application as filed in accordance with Art. 34 (2)(b) PCT.

Amendments of the claims

The independent claims 1, 15 and 23 have all been amended so that it is included that the invention relates to the field of *on-line analytical processing*, cf. the application page 1: *Field of the invention*. These amendments are in accordance with Item VIII 1.1. of the Written Opinion and should clarify the meaning of the terms used in the claims to be the specific meaning of the terms within the technical field. Thus, referring to text books within the technical fields such as: *Ralph Kimball: The Data Warehouse Toolkit, Wiley & Sons, 1996*, the meaning of the terms

dimension value
strict
aggregate

is clear.

It is in claim 1 added that an "at least partly aggregation normalised dimension" is a dimension with improved summarisability, cf. page 3, line 4-5, and that "irregularities of the dimension" are features rendering the dimension non-summarisable.

The structure of claims 2 and 3 is changed to avoid the reference to "the execution". Definitions of "covering", "onto" and "strict" are added to claim 2 based on the description page 2, lines 18-20.

Claims 13 and 14 have been rearranged to formulate the technical steps to be performed in a more clear way.

The terms "aggreation queries" and "navigation queries" have been defined in claim 19 based on the description page 38, line 10-12.

Amendments of the description

A discussion of D1 is added on the enclosed page 2a in accordance with Item VII and a list of the figures of the application is added on the enclosed page 13a.

References to the figures in the detailed part of the description was partly erroneous and corrections are added on the enclosed pages 24, 28, 31, 36, 37, 38, 39, 41 and 42.

All amendments are firmly based on the on the application as filed in accordance with Art. 34 (2)(b) PCT.

In case the Examiner does not agree that the new claims are properly based on the documents originally filed, a telephone interview with the Examiner pursuant to Rule 66.6 PCT is requested prior to the issuance of a preliminary examination report.

Yours sincerely,

Plougmann, Vingtoft & Partners

JJS

Jens Jørgen Schmidt

Form 1037
Set of amended claims
Set of amended claims with the amendments indicated (by confirmation copy only)
New description pages 2, 2a, 13, 13a, 24, 28, 31, 36, 37, 38, 39, 41 and 42

(iid) for each dimensional value of the child category, performing the steps of: creating a new dimension value of the new fused category representing the one or more values of the parent category to which the dimensional value of the child category is linked and creating links from said new dimension value to said values in the parent category, the creation of the new dimension value being conditioned that no dimension value of the new fused category already exists having exactly such link(s), and

for each category being a direct predecessor of the parent category of which category at least one dimension value of the parent category is linked to a dimension value of, setting said category as a grandparent category and creating links from the new dimension value to the one or more dimension values of the grandparent category to which said one or more dimensional values of the parent category are linked,

(iie) removing the links from the parent category to the one or more grandparent categories, whereby the grandparent categories no longer are direct predecessors of the parent category,

(iif) creating links from each dimensional value of the child category to the dimension value of the new fused category having the same links to the dimension values of the parent category whereby the new fused category becomes a direct predecessor of the child category, and removing the links from the dimension values of the child category to the parent category, whereby the parent category no longer is a direct predecessor of the child category,

and

(iig) setting the new fused category as the child category and returning to step (ii).

7. A method according to any of claims 3-6, wherein the make-onto procedure comprises the steps of, starting from the top category and successively proceeding towards the bottom category,

creating, for each dimension value of each category above the bottom category not being linked to any dimensional value of the category immediately below, a new dimension value in the category immediately below and creating a link between said new dimension value and said dimension value of the category in question.

8. A method according to any of claims 3-7, wherein the make-onto procedure comprises the successive steps of

(i) setting the top category of the dimension as the parent category,

(ii) for each category immediately below the parent category and having dimension values being linked to dimension values of the parent category, setting said category as the child category and perform the steps of

(iia) creating, for each dimension value of the parent category not being linked to any dimensional value of the child category, a new dimension value in the child category and creating a link between said new dimension value and said dimension value of the parent category,

(iib) setting the child category as parent category,

(iic) ending the make-onto procedure in case the parent category is the bottom category of the dimension, else returning to step (ii) of the make-onto procedure.

9. A method according to any of claims 4-8, wherein the make-covering procedure comprises the successive steps of

identifying links between dimension values of two categories having at least one intermediate category there between,

creating a new dimension value in each of said intermediate categories for each of those links for which no paths of links exists going only through immediate child-parent links from lower to higher categories and including a link to a dimension value of the intermediate category, and

replacing those links with links between the dimension values of those links and the new dimension values.

10. A method according to any of claims 4-9, wherein the make-covering procedure comprises the successive steps of

(i) setting the bottom category of the dimension as the child category,

(ii) for each category immediately above the child category for which at least one link between a dimension value of said category and a dimension value of the child category exists, setting the category as the parent category and perform the steps of:

(iia) ending the make-covering procedure for the parent category in case the parent category is the top category of the dimension:,

(iib) for each higher category being a direct predecessor category of the child category and being higher in the hierarchy than the parent category, performing the steps of

(iiba) identifying sets of dimension values of the higher category and dimension values of the child category for which sets

a link exists, and

no paths of links going only from lower to higher categories and including a link to a dimension value of the parent category exists, and

(iibb) creating for each identified set of dimension values a new dimension value in the parent category, creating links between each of the dimension values of the set and the new dimension value, and removing the link between the two dimension values of the identified set, whereby the higher category no longer is a predecessor of the child category,

(iic) setting the parent category as the child category and returning to step (ii).

11. A method for by means of a computer to at least partly aggregation normalise a multidimensional object including a set of facts comprising a plurality of facts mapped on a plurality of dimensions having dimension values organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension, by means of applying the method of any of claims 1-10 to at least one of the dimensions of the multidimensional object.

12. A method according to claim 11, wherein the multidimensional object comprises a plurality of facts and the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, the facts constituting the bottom layer of each of the dimensions of the multidimensional object.

13. A method according to claim 11 or 12, comprising the steps of

selecting a subset of categories of the one or more dimension to be aggregation normalised, and

performing an aggregation normalisation of the selected subset,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised.

14. A method according to any of claims 11-13, comprising the steps of

selecting specific aggregation functions to be performed on the multidimensional object, and

selecting by means of the computer normalisation steps to be performed based on the selection of specific aggregation functions to be performed,

whereby one or more of the dimensions of the multidimensional object is/are only partly aggregation normalised. .

15. A method for by means of a computer to at least partly aggregation normalise a general on-line analytical processing multidimensional object including a set of facts comprising a plurality of facts mapped on an aggregation normalised plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering, the multidimensional object comprising mappings of links between dimension values within each dimension,

the method comprising the steps of

retrieve the mapping from data storage means associated with the computer, including the mapping of the plurality of facts on the multidimensional object into the mapping of the multidimensional object so that the mapping comprises links from each of the facts to at least one dimension value in each of the plurality of dimensions, and the facts constitutes the bottom layer of each of the dimensions of the multidimensional object,

analysing the mapping of the multidimensional object to determine irregularities of the dimensions by means of analysing means executed by the computer,

creating new dimension values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is at least partly aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

16. A method according to claim 15, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-strict procedure for making the multidimensional object aggregation strict, thereby making the non-strict multidimensional object aggregation normalised, the make-strict procedure being executed on the condition that the multidimensional object is covering prior to the execution.

17. A method according to claim 15 or 16, wherein the step of creating new dimensional values and modifying the mapping comprises the step of

executing a make-covering procedure for making the multidimensional object covering, thereby at least partly making the non-covering multidimensional object aggregation normalised.

18. A method according to any of claims 15-17, wherein the method comprises the initial step of making each of the plurality of dimensions aggregation normalised by means of the method according to any of claims 1-10.

19. A method according to any of claims 11-18, wherein the created new dimensional values are marked as such, a pre-aggregation is performed on a multidimensional object being normalised by means of the computer and the method further comprises the step of

producing a reply to a query made to the system and concerning the multidimensional object, aggregate queries, exploring the dimension hierarchies, as well as navigation queries, that summarise the data at various levels of detail, in which reply the existence of the created new dimensional values is transparent.

20. A method according to any of claims 11-19, further comprising the steps of

implementing, into the aggregation normalised multidimensional object, of new facts including mapping of the facts onto the dimension, of new dimension values of the dimensions, or of new mapping between some of the dimension values, by which implementation irregularities of the multidimensional object is introduced,

analysing the introduced irregularities of the dimensions of the multidimensional object,

creating new dimensional values of the multidimensional object and modifying the mapping between dimensional values of the multidimensional object according to the analysis, whereby the multidimensional object is aggregation normalised, and

saving the new dimensions and the modified mapping in data storage means of the computer.

21. A computer system comprising at least one general purpose computer having data storage means associated therewith on which data storage means is stored a computer programme product suitable for adapting the computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20, the computer system comprising means for retrieving the computer programme product and perform accordingly.

22. A computer programme product suitable for adapting a general purpose computer to perform an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

23. A computer system for on-line analytical processing having data storage means associated therewith on which a multidimensional object is stored, the multidimensional object including

a set of facts comprising a plurality of facts,

a first plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a first mapping of links between dimension values within each dimension of the first plurality of dimensions as well as links between the facts and the dimensions of the first plurality of dimensions, at least one of the dimensions of the first plurality of dimensions being irregular, and

a second plurality of dimensions having dimension values being organised into categories of dimension values based on a partial ordering and comprising a second mapping of links between dimension values within each dimension of the second plurality of dimensions as well as links between the facts and the dimensions of the second plurality of dimensions, each of the second plurality of dimensions being aggregation normalised,

the computer system comprising a query handler component being adapted for producing replies to queries made to the computer system and concerning the multidimensional object, the replies to navigation queries being based on the first set of dimensions and the replies to aggregate queries being based on the second set of dimensions.

24. A computer system according to claim 23, wherein a set of pre-aggregation data relating to the second plurality of dimensions is further stored within the data storage means and the replies to aggregate queries furthermore are based on the set of pre-aggregation data.

25. A computer system according to claim 23 or 24, wherein the query handler component is adapted for producing replies to aggregate queries in which replies the existence of the second plurality of dimensions is transparent.

26. A computer system according to claim 25, wherein the query handler component is adapted for transforming aggregate queries made to the first plurality of dimensions into queries for the second set of dimensions and transforming replies based on the second set of dimensions into replies as based on the first set of dimensions, thus making the existence of the second plurality of dimensions transparent in the produced reply.

27. A computer system according to claim 26, wherein the multidimensional object is stored within the data storage means of the computer system in tables organised as a combination of star schemes for the part of the multidimensional object containing only strict mappings, and additional tables containing the non-strict part of the mappings, the query handler component makes use of said tables in transforming queries and replies.

28. A computer system according to any of claims 23-27 further comprising means adapted for performing an at least partly aggregation normalisation of a multidimensional object according to the method of any of claims 11-20.

INTERNATIONAL SEARCH REPORT

Intern. Application No
PCT/DK 00/00354

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JIM GRAY ET AL: "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals" DATA MINING AND KNOWLEDGE DISCOVERY 1, 1997, pages 29-53, XP002901286 page 40	1,11-15, 18,20-22
A	---	2-10,16, 17,19, 23-28
A	H.J. LENZ ET AL: "Summarizability in OLAP and Statistical Data bases" SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT, 1997. PROCEEDINGS, NINTH INTERNATIONAL CONFERENCE , pages 132-143, XP002901287 the whole document	1-28
	--- -/--	

[X] Further documents are listed in the continuation of box C. [ ] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 October 2000	24. 11. 2000

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Oskar Pihlgren

2

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/DK 00/00354

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	INDERPAL SINGH MUMICK: "Maintenance of Data Cubes and Summary Tables in a Warehouse" AT&T LABORATORIES, [Online] 1997, XP002901288 Retrieved from the Internet: <URL:http://citeseer.nj.nec.com/did/38362; > [retrieved on 2000-10-06] the whole document -----	1-28

2

Form PCT/ISA/210 (continuation of second sheet) (July 1992)